

ELECTRÓNICA

NUEVA

AUTO SWITCH para evitar
CORTOCIRCUITOS en la red

ANEMÓMETRO
PROGRAMABLE



TÉSTER para
PUERTO PARALELO

PROGRAMACIÓN con
microcontroladores ST7 LITE 09

LOS MATAJES MÁS POPULARES

DETECTOR de ULTRASONIDOS
CODIFICADOR FM ESTÉREO



rte

DIRECCIÓN
C/ Ferraz, 37
Teléf: (91) 542 73 80
Fax: (91) 559 94 17
MADRID 28008

DIRECTOR EDITORIAL:
Eugenio Páez Martín

Diseño Gráfico:
Paloma López Durán

Redactor:
Roberto Quirós García

SERVICIO TÉCNICO
Lunes y Miércoles de 17 a 20 h.
Teléf.: 91 542 73 80
Fax: 91 559 94 17
Correo Electrónico:
tecnico@nuevaelectronica.com

SUSCRIPCIONES
CONSULTAS
PEDIDOS
Teléf.: 91 542 73 80
Fax: 91 559 94 17
Correo Electrónico:
comercial@nuevaelectronica.com

PAGINA WEB:
www.nuevaelectronica.com

FOTOMECÁNICA:
Videlec S.L.
Teléf.:(91) 375 02 70

IMPRESIÓN:
IBERGRAPHI 2002
C/ Mar Tirreno 7
San Fernando de Henares - Madrid

DISTRIBUCIÓN:
Coedis, S.A.
Teléf.:(93) 680 03 60
MOLINS DE REI
(Barcelona)

Traducción en Lengua
española de la revista
"Nuova Elettronica", Italia.
DIRECTOR GENERAL
Montuschi Giuseppe

DEPÓSITO LEGAL:
M-18437-1983

Suscripción anual	45,60 Euros
Susc. certificada	70,60 Euros.
Europa	73,80 Euros.
América	143,00 Euros.

Cupón de suscripciones y pedidos en
página 37.

Nº 241
4,80 Euros. (Incluido I.V.A.)
Canarias, Ceuta y Melilla
4,80 Euros (Incluidos portes)

www.nuevaelectronica.com

SUMARIO

En este número

TÉSTER para PUERTO PARALELO



Muchos circuitos electrónicos pueden ser controlados por el puerto paralelo del ordenador, conexión que normalmente utilizamos para la impresora. En este artículo, además de presentar un interesante dispositivo complementado con varias aplicaciones, aclaramos todos los conceptos relacionados con el funcionamiento de este puerto de comunicaciones.

(LX.1588)pag.4

PROGRAMACIÓN con microcontroladores ST7 LITE 09 (5)

Una de las peculiaridades del lenguaje Assembler para los micros ST7 es la posibilidad de direccionar los operandos de una instrucción con varios modos diferentes. Hoy vamos a aprender a reconocer y a utilizar los modos de direccionamiento indirectos y los modos indexados indirectos.pag.24

CUPÓN DE PEDIDOS Y SUSCRIPCIONESpag.37

AUTO SWITCH para evitar CORTOCIRCUITOS en la red



Como en algunas ocasiones este proyecto ha sido desarrollado como consecuencia de vuestras peticiones y sugerencias. En concreto, algunos lectores nos han preguntado sobre dispositivos para ahorrar en la factura mensual de la compañía eléctrica y por sistemas de protección eléctrica para los dispositivos que disponen de la función "stand by".

(LX.1589)pag.42

ANEMÓMETRO PROGRAMABLE



Este anemómetro puede ser programado para excitar un relé o para emitir una señal acústica cuando el viento supera una determinada velocidad que podemos ajustar. El relé puede utilizarse para controlar una sirena, un motor o cualquier otro dispositivo eléctrico.

(LX.1606)pag.52

CATÁLOGO DE KITSpag.60

LOS MONTAJES MÁS POPULARES

DETECTOR de ULTRASONIDOS

Casi todo el mundo sabe que los murciélagos utilizan un radar ultrasónico para identificar objetos en la oscuridad y también que los perros son capaces de captar los ultrasonidos. ¿Nunca os habéis preguntado si un ser humano puede oírlos? La respuesta a esta pregunta la ofrece este proyecto.

(LX.1428)pag.63

CODIFICADOR FM ESTEREO

Conectado la salida de este Codificador FM Estéreo (Stereo FM Encoder) a la entrada de cualquier Transmisor FM Mono pasará a transmitir en estéreo la señal estereofónica presente en la entrada del Codificador.

(LX.1226)pag.67



TÉSTER para

Muchos circuitos electrónicos pueden ser controlados por el puerto paralelo del ordenador, conexión que normalmente utilizamos para la impresora. En este artículo, además de presentar un interesante dispositivo complementado con varias aplicaciones, aclaramos todos los conceptos relacionados con el funcionamiento de este puerto de comunicaciones.

En los ordenadores de última generación casi todas las conexiones con los periféricos, como por ejemplo la conexión a una impresora, se desarrollan a través de puertos serie tipo USB (acrónimo de Universal Serial Bus).

Los puertos **USB** están caracterizados por una elevada velocidad de transferencia, por permitir la conexión de varios periféricos (ratón, módem, teclado, etc.) y por poder realizar conexiones en caliente, es decir con el ordenador encendido. Aunque esta sea la tendencia actual, la mayoría de los ordenadores disponen de un **puerto paralelo**.

Como generalmente a este puerto del ordenador se conecta el cable que va a la **impresora**, la mayoría de la gente piensa que su única función es precisamente esta.

Esto no es así, se trata de un puerto de comunicaciones vía paralelo que puede utilizarse como **canal de entrada-salida** al ordenador. Con el uso de una **interfaz** adecuada, como la que aquí presentamos, se pueden realizar múltiples aplicaciones: Control de **temperatura**, medición de **sensores**, control de **luces**, control de **motores paso-a-paso**, adquisición de **datos**, etc.

EL PUERTO PARALELO

Para localizar el conector del puerto paralelo del ordenador, hay que buscarlo en el panel trasero de conexiones: Se trata de un conector **hembra de 25 pines** tipo **SUB-D**. En este conector de **25 terminales** hay **17 señales útiles**, ya que las **8 restantes** se conectan a masa.

En la mayoría de los ordenadores de última generación se suele utilizar un color diferente para cada conector: Para el **puerto paralelo** se utiliza el color **fucsia**, mientras que el conector del **ratón PS/2** es **verde claro**, el conector del **teclado** es **violeta** y los conectores de los **puertos serie** son **verde oscuro**.

Cuando se creó el puerto paralelo el juego de **caracteres ASCII** imprimibles era limitado debido a que las impresoras utilizaban un juego de caracteres muy parecido a los de los teletipos. Con ocho señales binarias en **paralelo, 8 bits**, se podían realizar las **256**

puerto paralelo y muchas impresoras llevan un conector desarrollado por Centronics para tal fin.

EPP (Enhanced Parallel Port): Con este estándar se utiliza el puerto paralelo de forma **bidireccional**, es decir puede, alternativamente, mandar **8 bits** simultáneos hacia la impresora o recibirlos. Además es unas **4 veces** más **rápido** que el **SPP**.

ECP (Extended Capabilities Port): Puerto bidireccional que trabaja con una gran tasa de transferencia ya que se apoya en el uso de **DMA (Acceso Directo a Memoria)**.

Tanto **EPP** como **ECP** son compatibles con **Centronics**.

Hoy en día los puertos paralelo incluidos en los ordenadores disponen de los **tres modos** operativos o, en su defecto, de los modos avanzados **EPP+ECP**. Solo los ordenadores **muy antiguos** disponen únicamente de puertos paralelo **SPP**.

puerto paralelo

combinaciones que determinaban los caracteres que, por aquellos tiempos, eran más que suficientes. La transmisión entre la impresora y el ordenador se realizaba de **4 en 4 bits**, es decir un **nibble** cada vez.

Con la aparición de los sistemas gráficos las impresoras evolucionaron muchísimo y, como consecuencia, su puerto de conexión natural por ese entonces, es decir el puerto paralelo, desarrollándose **sistemas más sofisticados**, dando lugar así a diferentes **tipos** de puerto paralelo, que al disponer del **mismo conector** se les suele denominar como diferentes **modos** operativos:

SPP (Standar Parallel Port): Modo de comunicación **original** desarrollado por **IBM** y **Centronics**, empresa que logró estandarizar este tipo de conexión. De hecho hoy se sigue utilizando el termino Centronics para referirse al

Determinar el TIPO de PUERTO

Para conocer el tipo de puerto paralelo del ordenador se puede hacer a través del **sistema operativo Windows** (siguiendo las instrucciones indicadas en las Figs.1 a 4) o a través del **Setup** de la **BIOS**, donde además de **ver** el modo operativo se puede **seleccionar** el modo deseado, como se detalla en el epígrafe siguiente.

Dado que **nuestra interfaz** está desarrollada para desempeñar funciones de **entrada y salida**, el puerto ha de estar en modo **EPP** o **ECP** para funcionar correctamente. Si el puerto paralelo está en modo **SPP** hay que **cambiarlo**, mediante el **Setup** de la **BIOS** del ordenador, con el procedimiento que indicamos a continuación.

Configuración del BIOS SETUP

Ya que no todo el mundo está familiarizado con la **BIOS** (tema tratado exhaustivamente en los artículos de **Nueva Informática**) presentamos

a continuación sus principios básicos de funcionamiento. **BIOS**, acrónimo de **B**asic **I**nput **O**utput **S**ystem, es el nombre que se da a una serie programas y datos almacenados en una memoria EPROM o en una Memoria Flash alojada en la **placa base** del ordenador.

Estos programas incluyen las **rutinas** de control de **dispositivos básicos** (teclado, disquetera, disco duro y puertos), el programa de **arranque del PC** y el programa de **configuración** de los recursos del ordenador (**Setup**). Por este motivo solo ha de acceder al **Setup** de la **BIOS** quien sepa bien lo que está

haciendo, ya que una incorrecta configuración puede provocar que no funcione algún dispositivo o, incluso, que **no** arranque el ordenador.

En principio las BIOS son muy diferentes, dado la gran variedad existente de placas base de ordenador. No obstante existen dos compañías especializadas que fabrican la gran mayoría de la BIOS para PC: **AMI** y **AWARD**. En este caso el procedimiento de configuración es bastante sistemático y claro, como describimos en los artículos correspondientes de **Nueva Informática**, y que recordamos brevemente a

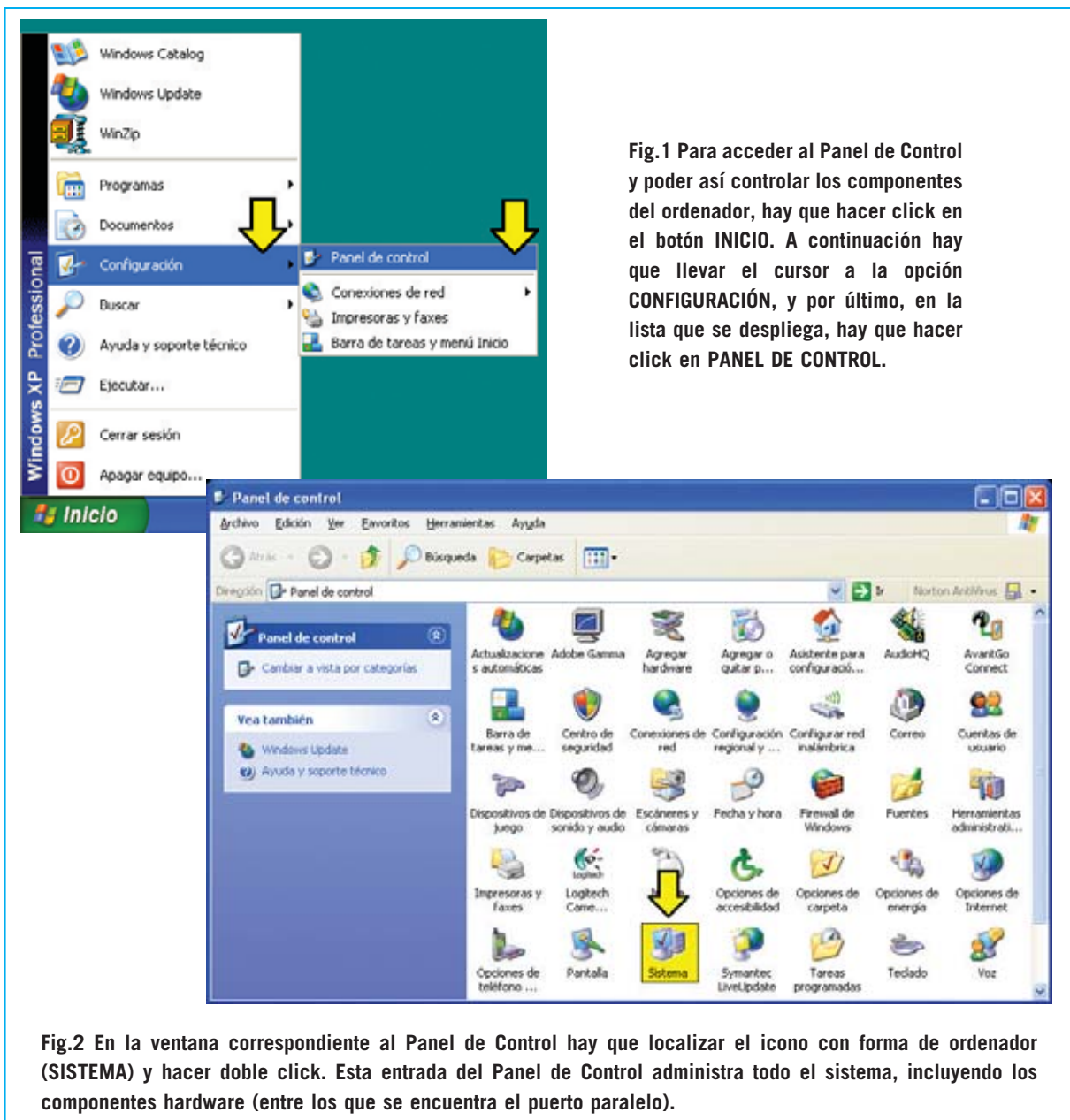


Fig.1 Para acceder al Panel de Control y poder así controlar los componentes del ordenador, hay que hacer click en el botón INICIO. A continuación hay que llevar el cursor a la opción CONFIGURACIÓN, y por último, en la lista que se despliega, hay que hacer click en PANEL DE CONTROL.

Fig.2 En la ventana correspondiente al Panel de Control hay que localizar el icono con forma de ordenador (SISTEMA) y hacer doble click. Esta entrada del Panel de Control administra todo el sistema, incluyendo los componentes hardware (entre los que se encuentra el puerto paralelo).

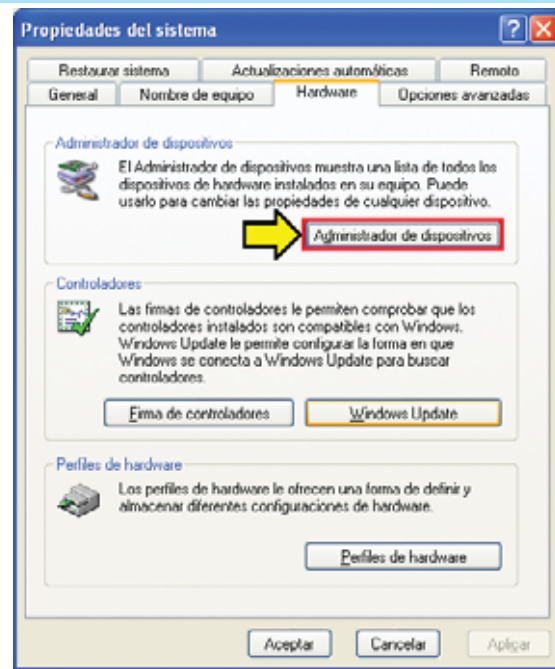


Fig.3 La ventana SISTEMA dispone de varias entradas diferentes a las que se accede a través de las pestañas superiores. Para acceder a la administración de los dispositivos hay que seleccionar la pestaña HARDWARE y, a continuación, el botón ADMINISTRADOR DE DISPOSITIVOS.

Fig.4 Para acceder a los puertos paralelo hay que hacer doble click sobre la línea PUERTOS (COM Y LPT) o hacer click en el signo (+) situado al lado. Automáticamente se despliega una lista con los puertos y su tipo, que en el caso de los puertos paralelo pueden ser: SPP (en este caso el sistema indica Puerto de impresora), EPP (en este caso el sistema indica Puerto de impresora EPP) o ECP (en este caso el sistema indica puerto de impresora ECP).

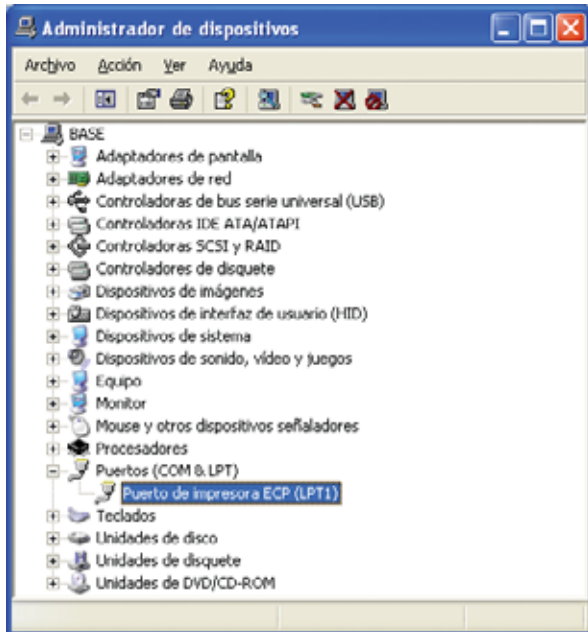


Fig.5 Aspecto del menú INTEGRATED PERIPHERALS en el Setup de las BIOS AMI y AWARD. La opción correspondiente al ajuste del modo del puerto (PARALLEL PORT MODE) está resaltada.

continuación. En el caso de **no** disponer de una BIOS AMI o AWARD hay que consultar el **manual** de la **placa base** para realizar esta tarea.

El **acceso** al **Setup** solo se puede hacer cuando el ordenador está **arrancando**, de hecho durante un momento se nos presenta en pantalla un mensaje similar a este: **Press DEL to Enter Setup**, que quiere decir **Presione la tecla Supr para entrar en el Setup**. Si pulsamos la tecla **Supr** (Suprimir, **DEL** en un teclado anglosajón) mientras está presente este mensaje el ordenador responde entrando en la **Pantalla principal** del **Setup** de la BIOS. Generalmente se utilizan las teclas de los **cursores** para el desplazarse por esta pantalla y la tecla **ENTER** para seleccionar un menú o función. Hay que localizar en pantalla el menú dedicado al ajuste de puertos y dispositivos, menú que en las **BIOS AWARD** y **AMI** se denomina **Integrated Peripherals**. Aquí hay que localizar el modo operativo del puerto paralelo (ver Fig.5), es decir la opción **Parallel Port Mode**.

Si al lado de esta opción pone **Normal** o **SPP**, hay que cambiarlo a **ECP** o **EPP**. Una vez hecho este cambio hay que volver al **menú principal** del Setup (pulsando la tecla **Esc**) y salir grabando los cambios (**Save and Exit Setup**). Con esta operación se ha **configurado** el puerto paralelo para trabajar en uno de los modos avanzados, **necesario** para nuestra interfaz **LX.1588**.

NOTA: Si hemos cambiado el modo, es posible que, si utilizamos alguna versión **antigua** de **Microsoft Windows**, nos pida el **CD ROM** del sistema operativo para copiar los programas de control que respondan al nuevo modo operativo del puerto paralelo.

SEÑALES del PUERTO PARALELO

Actualmente el puerto paralelo tiene la configuración eléctrica mostrada en la Tabla N.1.

Puesto que el **puerto paralelo** se diseñó para conectar impresoras al ordenador, el

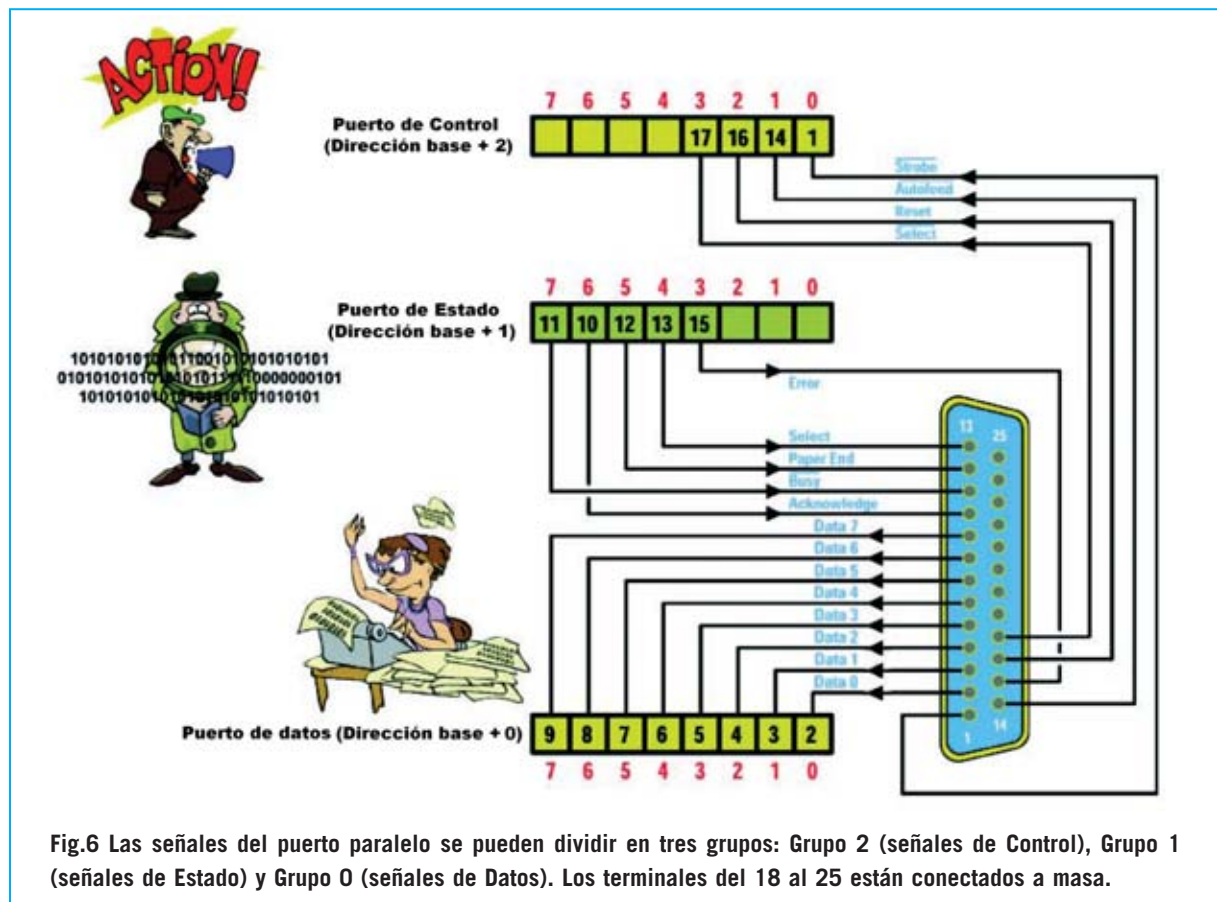


Fig.6 Las señales del puerto paralelo se pueden dividir en tres grupos: Grupo 2 (señales de Control), Grupo 1 (señales de Estado) y Grupo 0 (señales de Datos). Los terminales del 18 al 25 están conectados a masa.

significado de las **señales** tiene que ver con el funcionamiento de las **impresoras**.

Las **señales** del puerto paralelo se pueden dividir en **tres grupos** (ver Fig.6).

El **Grupo 0** corresponde a las señales que van desde el terminal **2** hasta el terminal **9**, es decir las **señales de datos (Data)**. Por estos terminales circula la información a imprimir en forma de datos binarios de **8 bits**.

El **Grupo 1** corresponde a las señales de **entrada** que llegan desde la impresora. Se trata de **5 señales** de entrada que avisan cuando se acaba el papel o bien si la escritura de una línea no se ha completado todavía. Además solicitan datos si la impresión ha finalizado o generan un error si, por ejemplo, se ha acabado la tinta. En la Fig.6 se muestran estos terminales (**10, 11, 12, 13 y 15**).

TABLA N.1

Grupo 0

Terminal	Señal	Configuración
2	Data 0	salida del PC
3	Data 1	salida del PC
4	Data 2	salida del PC
5	Data 3	salida del PC
6	Data 4	salida del PC
7	Data 5	salida del PC
8	Data 6	salida del PC
9	Data 7	salida del PC

Grupo 1

Terminal	Señal	Configuración
10	Acknowledge	entrada al PC
11	Busy	entrada al PC
12	Paper End	entrada al PC
13	Select	entrada al PC
15	Error	entrada al PC

Grupo 2

Terminal	Señal	Configuración
1	Strobe	salida del PC
14	Autofeed	salida del PC
16	Reset	salida del PC
17	Select	salida del PC

El **Grupo 2** corresponde a las señales de los terminales **1-14-16-17**. Se trata de **4 señales de salida** para el control y sincronización de la mecánica de la impresora.

Los terminales **18, 19, 20, 21, 22, 23, 24 y 25 (GND)** van conectados a **masa**.

Esta división de tres grupos es una división **funcional**, es decir cada grupo tiene una función diferente, si bien las señales **no** se separan en tres conjuntos de cables (buses) diferentes. En un **mismo** conjunto de cables (**bus**) se introducen todas las señales, si bien los tres grupos corresponden a tres **direcciones diferentes**.

Cuando se dice que el puerto paralelo trabaja a **8 bits** se hace referencia a que la **transmisión de datos (data)** se hace con **8 líneas simultáneas** (terminales **2 a 9** denominados funcionalmente **Data0, Data1 ... Data7**).

¿Por qué TRES GRUPOS?

El motivo principal de la agrupación de las señales del puerto paralelo es de carácter muy práctico: Distribuyendo las señales en tres grupos (de un **byte**) es más fácil manipular los valores. Por ejemplo, para poner a **nivel lógico 1** únicamente los terminales **4, 3 y 9**, hay que formar el número binario:

Pin	9	8	7	6	5	4	3	2
Bit	1	0	0	0	0	1	1	0

Observando los **Grupos 1 y 2**, se puede apreciar que en algunas señales hay una **línea encima** del nombre de la señal. Esta línea significa que el estado lógico de la señal está **invertido**, es decir que trabaja con lógica **inversa**. Expresado de una forma más clara, estas señales se **ACTIVAN** con un nivel **lógico 0** en lugar de con un nivel lógico 1.

Para entender este comportamiento, muy común en algunos circuitos electrónicos, pero que no todo el mundo conoce, mostramos dos ejemplos con el **Grupo 2**, que tiene las **señales de los terminales 1, 14 y 17 invertidas**.

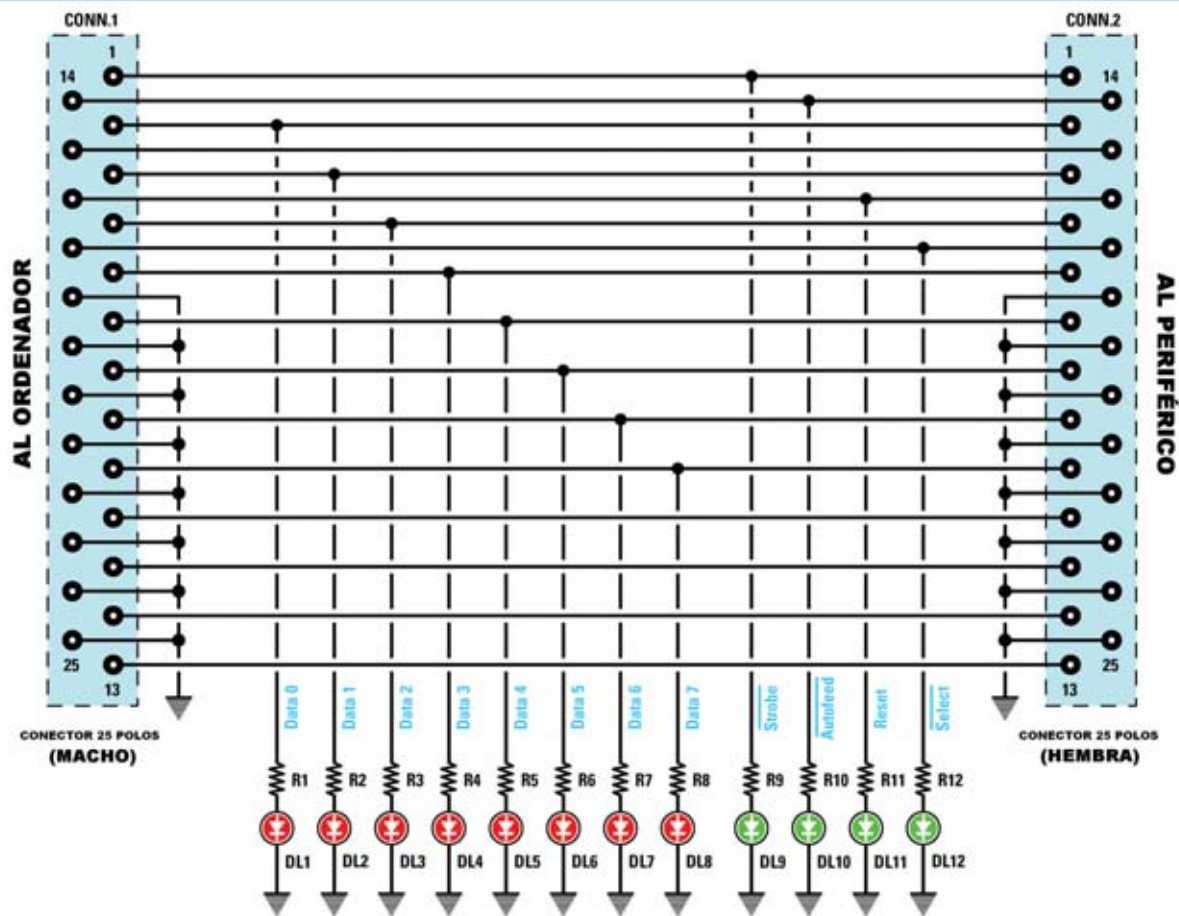


Fig.7 Esquema eléctrico del Téster LX.1588, útil para controlar en tiempo real el funcionamiento del puerto paralelo. Los 8 diodos LED rojos (DL1-DL8) están conectados al Grupo 0 para monitorizar las señales de datos, mientras que los 4 diodos LED verdes monitorizan las señales del Grupo 2 (Control).

LISTA DE COMPONENTES LX.1588

R1 = 1.000 ohmios 1/8 vatio
 R2 = 1.000 ohmios 1/8 vatio
 R3 = 1.000 ohmios 1/8 vatio
 R4 = 1.000 ohmios 1/8 vatio
 R5 = 1.000 ohmios 1/8 vatio
 R6 = 1.000 ohmios 1/8 vatio
 R7 = 1.000 ohmios 1/8 vatio
 R8 = 1.000 ohmios 1/8 vatio
 R9 = 1.000 ohmios 1/8 vatio
 R10 = 1.000 ohmios 1/8 vatio

R11 = 1.000 ohmios 1/8 vatio
 R12 = 1.000 ohmios 1/8 vatio
 DL1 = Diodo LED rojo
 DL2 = Diodo LED rojo
 DL3 = Diodo LED rojo
 DL4 = Diodo LED rojo
 DL5 = Diodo LED rojo
 DL6 = Diodo LED rojo
 DL7 = Diodo LED rojo
 DL8 = Diodo LED rojo

DL9 = Diodo LED verde
 DL10 = Diodo LED verde
 DL11 = Diodo LED verde
 DL12 = Diodo LED verde
 CONN.1 = Conector SUB-D 25 macho
 CONN.2 = conector SUB-D 25 hembra

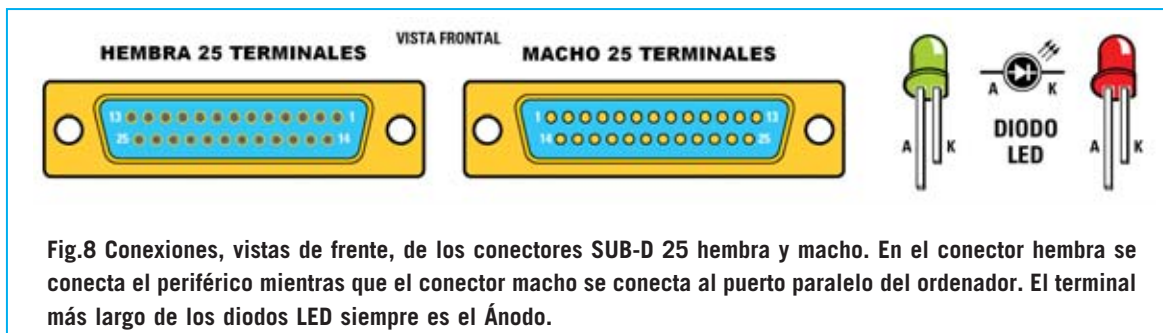


Fig.8 Conexiones, vistas de frente, de los conectores SUB-D 25 hembra y macho. En el conector hembra se conecta el periférico mientras que el conector macho se conecta al puerto paralelo del ordenador. El terminal más largo de los diodos LED siempre es el Ánodo.

Para tener en salida el estado lógico **1011** hay que escribir:

Pin					17	16	14	1
Seg					SeI	Res	Aut	Str
Bit					0	0	0	0

Para tener en salida el estado lógico **0101** hay que escribir:

Pin					17	16	14	1
Seg					SeI	Res	Aut	Str
Bit					1	1	1	0

Prestando un poco de atención no es un proceso complicado. Cuando hayáis instalado en vuestro ordenador el programa que acompaña al **Téster para puerto paralelo** se puede practicar con estas **señales**.

A continuación presentamos el esquema eléctrico y los detalles de montaje del **Téster**. Después mostramos como cada uno puede desarrollar sus propias aplicaciones, tanto en entorno DOS como en entorno Windows, y algunos esquemas que nosotros hemos desarrollado.

ESQUEMA ELÉCTRICO

Como se puede observar en la Fig.7 se trata de un esquema muy simple, constituido por **conectores, diodos LED** y **resistencias**.

En el **Grupo 0** hemos conectado **diodos LED** de color **rojo** y en el **Grupo 2** **diodos LED** de color **verde**. El **Grupo 1**, al tener señales de **entrada**, no puede ser activado desde el ordenador sino desde el periférico que está conectado al puerto paralelo del ordenador.

La tensión en las salidas del puerto paralelo tiene niveles **TTL**, es decir **5 voltios** para el **nivel lógico 1**. Por este motivo hemos conectado en **serie** a cada **diodo LED** una **resistencia** que **limita** la **corriente** al valor necesario para su iluminación, sin sobrecargar las salidas.

Como se puede ver en el esquema eléctrico, **todos** los terminales de los conectores macho y hembra de 25 polos están **conectados**. De

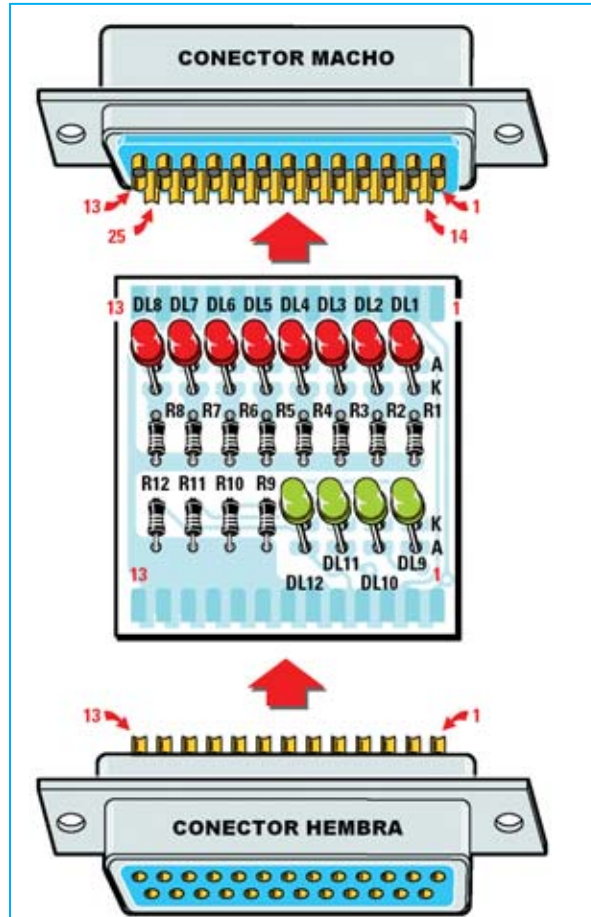


Fig.9 Esquema práctico de montaje del Téster para puerto paralelo LX.1588. Los diodos LED rojos se montan orientando el Ánodo hacia la parte superior mientras que los diodos LED verdes se montan orientando el Ánodo hacia la parte inferior.

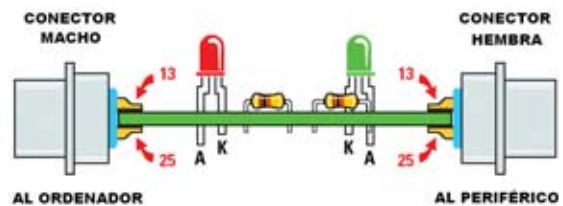


Fig.10 Como se puede observar el circuito impreso se aloja entre las dos filas de terminales de los conectores. Los terminales de los conectores se sueldan directamente a las pistas del circuito impreso.

esta forma podemos visualizar todas las señales en **tiempo real** a través de los diodos LED mientras el dispositivo está conectado al ordenador y funcionando.

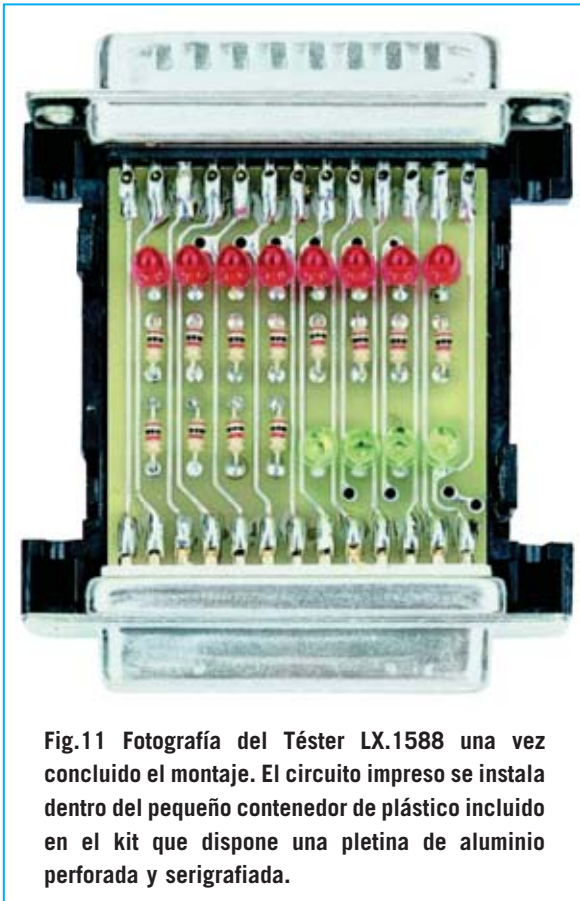


Fig.11 Fotografía del Téster LX.1588 una vez concluido el montaje. El circuito impreso se instala dentro del pequeño contenedor de plástico incluido en el kit que dispone una pletina de aluminio perforada y serigrafiada.

También podemos probar circuitos y programas de aplicación desarrollados para el puerto paralelo, como los que presentamos en este artículo.

REALIZACIÓN PRÁCTICA

Hemos desarrollado el circuito impreso de modo que se pueda alojar dentro de un minúsculo **contenedor** de plástico estándar, similar a los utilizados para los **Téster RS232**. Los laterales del circuito están perfilados para los conectores SUB-D 25 macho y SUB-D 25 hembra.

Aconsejamos comenzar el montaje por las **resistencias**, todas del mismo valor. Después de soldarlas hay que cortar con unos alicates la parte excedente de sus terminales de conexión.

A continuación hay que montar en el circuito impreso los **conectores** de tal modo que el circuito impreso quede alojado entre las dos filas de terminales de cada conector, orientándolos como se indica en las Figs.9-10.



Fig.12 El Téster LX.1588 se conecta al puerto paralelo del ordenador, bien directamente o a través de un cable alargador que incorpore conectores SUB-D 25 macho y hembra en sus extremos.



Fig.13 El primer paso para instalar el programa LPT Tester consiste en hacer click en el botón INICIO y, a continuación, hacer click en EJECTAR.

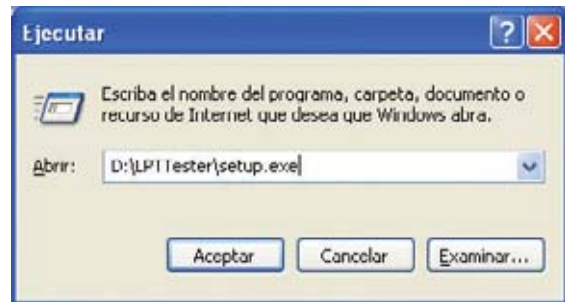


Fig.14 En el recuadro central hay que escribir D:\LPTTESTER\SETUP.EXE, confirmando la operación haciendo click en ACEPTAR. La letra D corresponde a la letra asignada a la unidad donde se ha introducido el CDROM.



Fig.15 Esta es la primera ventana que aparece en la instalación. Después de un momento desaparece y se muestra la ventana de la Fig.16.

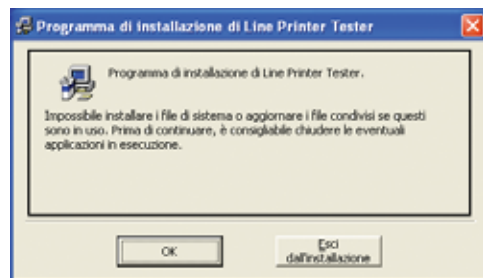


Fig.16 Para comenzar la instalación del programa Line Printer Tester hay que hacer click en el botón OK.

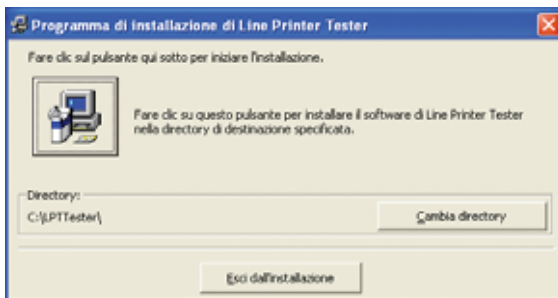


Fig.17 El directorio predeterminado para la instalación es C:\LPTTester\. Para continuar hay que hacer click en el icono con forma de ordenador.

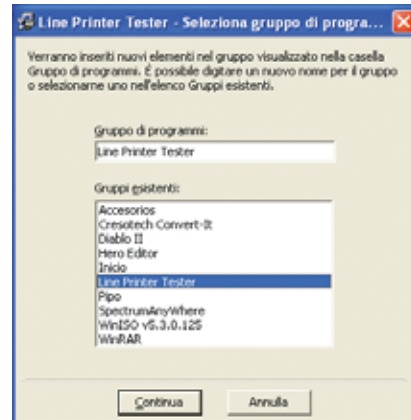


Fig.18 El software para el Téster LX.1588 se instala de forma predeterminada en el grupo PROGRAMAS. Para continuar hay que hacer click en el botón CONTINUA.

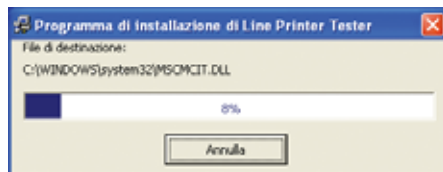


Fig.19 En la fase de copia de archivos se muestra una barra de progreso de color azul.

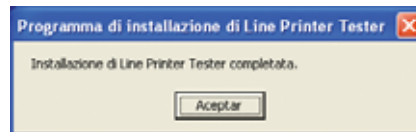


Fig.20 LA instalación del programa LPT Tester ha concluido. Para terminar hay que hacer click en ACEPTAR.

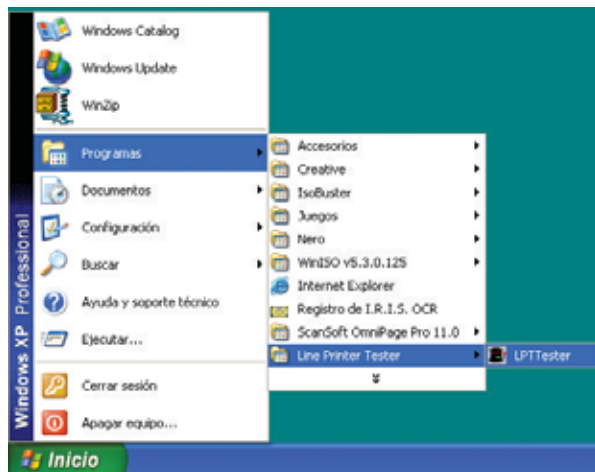


Fig.21 Para ejecutar el programa LPT Tester hay que hacer click en el botón INICIO. A continuación hay que llevar el cursor sobre PROGRAMAS y, en la lista desplegada, hay que localizar la entrada LINE PRINTER TESTER. Por último hay que hacer click sobre LPT TESTER.

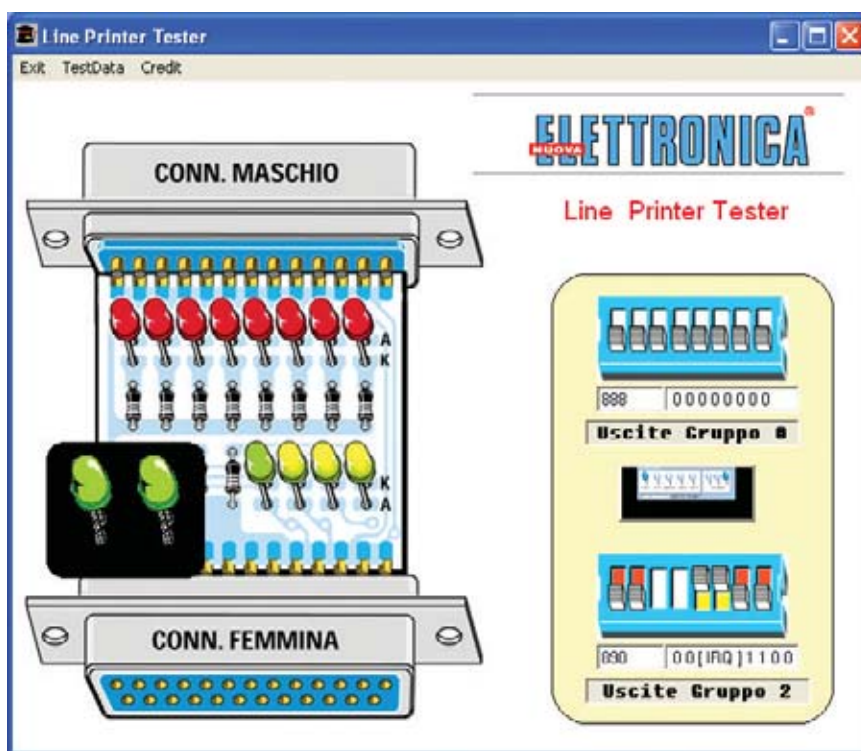
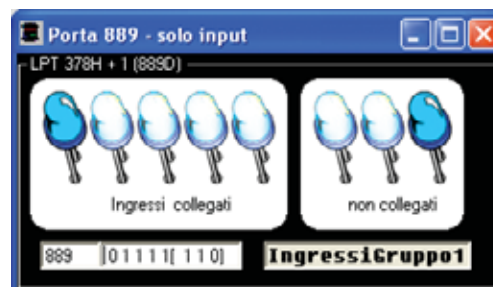


Fig.22 Así se presenta la ventana principal del programa LPT Tester. En la parte izquierda está representado el circuito impreso con sus diodos LED, mientras que en la parte derecha se encuentran unos dip-switch que controlan las señales del Grupo 0 y del Grupo 2. Actuando sobre estos dip-switch se controlan las salidas del puerto paralelo.

Fig.23 Al hacer click sobre el Grupo 1 (ver Fig.22) se abre esta ventana en la que se muestran los niveles lógicos de las señales de Estado del puerto paralelo.



Como se puede observar el **conector macho** se instala en la parte **superior** mientras que el **conector hembra** se instala en la parte **inferior**. Al soldar los terminales a las pistas del circuito impreso hay que tener cuidado en **no cortocircuitar** dos pistas adyacentes con un exceso de estaño.

Ahora hay que realizar los **agujeros** para los **diodos LED** en una de las **tapas** del **contenedor** de plástico, tomando como referencia la pletina perforada y serigrafiada. Para realizar los agujeros se puede utilizar un taladro con una broca de **3 milímetros**.

Llegado este punto se pueden introducir los **diodos LED** en sus agujeros. Los **ocho** diodos LED **rojos** corresponden a las referencias **DL1-DL8** mientras que los **cuatro** diodos LED **verdes** corresponden a las referencias **DL9-DL12**.

El terminal más largo de los diodos, el ánodo, debe insertarse en el agujero marcado con la letra **A**, que se encuentra orientado hacia **arriba** en los diodos LED **rojos** y orientado hacia **abajo** en los diodos LED **verdes**. Después de haber introducido los terminales de los diodos LED en sus agujeros hay que abrirlos hacia fuera sin soldarlos (de momento).

Antes de soldar los terminales hay que controlar la **altura** a la que han de quedar los diodos LED, superponiendo en el circuito impreso la **tapa** del contenedor correspondiente al lado de los diodos LED. Una vez controlada la altura correcta se pueden soldar los terminales y cortar con unos alicates la parte excedente de sus terminales de conexión.

Ya se puede por fin cerrar el contenedor y probar el dispositivo, conectándolo al puerto paralelo. Si hay una **impresora** conectada en el puerto, en primer lugar hay que desconectarla, a continuación conectar el **Téster** al puerto paralelo y conectar la impresora al **Téster**, quedando este entre medias del puerto y de la impresora, **señalizando** a través de los **diodos LED** el estado de las **señales**.

NOTA: Se puede utilizar un **cable prolongador** con conectores macho-hembra si no se desea conectar directamente el **Téster LX.1588** al puerto paralelo. Este cable es particularmente útil cuando el conector del puerto paralelo está en una zona poco accesible (ver conexión en la Fig.12).

SOFTWARE para el TÉSTER

Para **probar** el **Téster LX.1588** hemos desarrollado un programa que trabaja en entorno **Windows**. Este programa también puede ser utilizado como **patrón** para los que quieran **desarrollar** sus **aplicaciones** utilizando el **código fuente**.

Para instalar el software hay que introducir el **CD** en la unidad del ordenador y lanzar el **programa de instalación**. Las **Figs.13 a 20** muestran detalladamente el **procedimiento** de instalación del programa.

Para abrir el programa de gestión del puerto paralelo hay que comenzar haciendo click en el botón **Inicio** del Escritorio de Windows. A continuación hay que llevar el cursor sobre **Programas** y, en la lista que se despliega, hay que localizar la entrada **Line Printer Tester** y posicionar el cursor sobre ella. Por fin, en el último cuadro desplegado, hay que hacer click sobre **LPT Tester**. La secuencia de estas operaciones se puede ver en la Fig.21.

Observando la ventana principal del programa (ver Fig.22), se puede ver en el lado izquierdo el **esquema práctico** de nuestro circuito, mientras que a la derecha hemos reproducido los **Grupos 0-2** del puerto paralelo y unos **dip-switch** en correspondencia con los terminales de estos grupos.

Haciendo click con el ratón sobre las palancas de los **dip-switch** se pueden llevar al estado lógico deseado (**1** o **0**) las señales, simulando así los estados lógicos de las salidas del puerto paralelo. Simultáneamente los **diodos LED** se iluminan en concordancia con el nivel lógico de la señal correspondiente (hay que tener en cuenta que en el **Grupo 2** hay algunas señales **invertidas**).

NOTA: Cuando se hace click con el ratón sobre un diodo LED aparece sobreimpresionado el **nombre** de la **señal** asignada.

Haciendo click sobre el cuadro del **Grupo 1** se abre una ventana como la mostrada en la Fig.23, donde se pueden observar los valores de las **señales de entrada**.

Si está conectado el **Téster LX.1588** al puerto paralelo se puede observar como al actuar sobre los **dip-switch** del **programa** se encienden los **diodos LED** correspondientes en el **Téster**.

Si además hay una **impresora** conectada al Téster, actuando sobre los dip-switch del **Grupo 2** se escucharán sonidos producidos por la **mecánica** de la impresora, suceso normal ya que estas señales gobiernan la mecánica de la impresora.

En el caso de que en el Téster esté conectado un **circuito de aplicación** se pueden **probar** las diferentes combinaciones para **analizar su funcionamiento** e, incluso, **diseñar** un **programa de gestión** para el circuito.

DESARROLLO de APLICACIONES

Seguramente después de conocer el funcionamiento del puerto paralelo y de nuestro **Téster LX.1588** muchos de vosotros estaréis interesados en **desarrollar programas** y **circuitos** que utilicen el puerto paralelo. A continuación vamos a mostrar algunos ejemplos de aplicaciones con sus instrucciones en código fuente y comentadas. Estas instrucciones deben ser escritas en **Visual Basic** (Entorno **Windows**) o en **GW BASIC** (entorno **DOS**) para poder ver sus resultados de forma práctica.

Los programas han sido escritos teniendo en consideración que el puerto paralelo está configurado como **LPT1**, denominación común para la impresora 1, que corresponde a la **dirección** de puerto **378H** del ordenador. Este número hexadecimal corresponde al número **888** decimal.

Aunque es improbable, puede que vuestro PC esté configurado como **LPT2** o que disponga

de un **segundo** puerto paralelo, en este caso se corresponde a la dirección de puerto **278H**. También existe la posibilidad, ya que las **BIOS** para PC soportan hasta **tres puertos paralelo**, que exista un tercer puerto paralelo o que el puerto de la placa base esté configurado como **LPT3**, en este caso la dirección del puerto del ordenador es **3BCH**.

ENTORNO DOS

Para quienes trabajen con el sistema operativo DOS, y quieran realizar los ejemplos que proponemos, han de tener uno de los siguientes programas:

GWBasic (Olivetti)

Básica (IBM)

QuickBasic45 (Microsoft)

TurboBasic (Borland)

En entorno **DOS** es muy sencillo acceder vía software al puerto paralelo: Solo se necesitan **dos instrucciones** y conocimientos básicos de **cálculo binario**.

Para escribir estos pequeños programas en primer lugar hay que arrancar un ordenador con el sistema operativo DOS. Al encender el PC se presentará el prompt típico **C :\>**. También se puede utilizar el DOS de los sistemas operativos que **incluyen** o que **coexisten** con DOS, **Windows 3.1**, **Windows 95** y **Windows 98**, en los dos últimos arrancando en **modo DOS**.

En la línea de comandos, a continuación del prompt **C :\>** hay que escribir:

Si tenéis el programa **Gwbasic**: **GW BASIC** y pulsar **ENTER**.

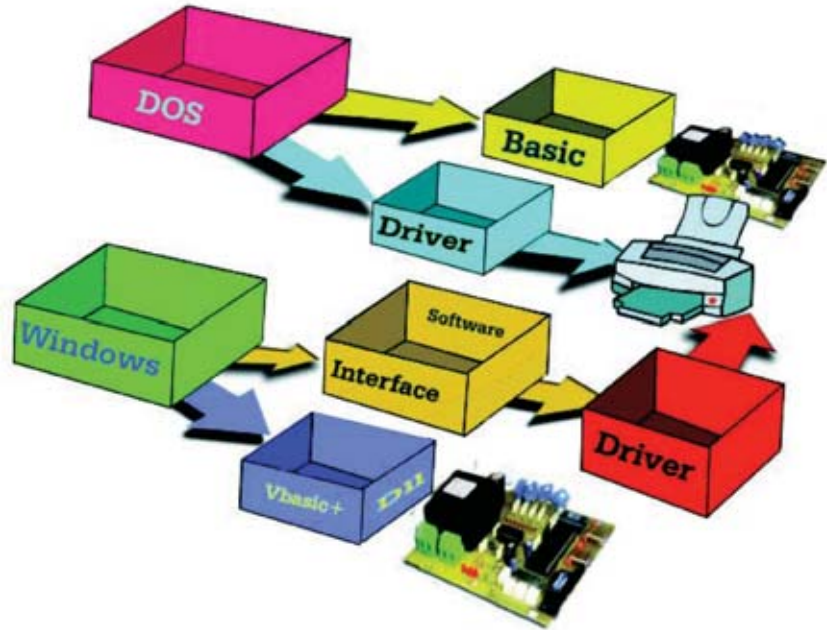
Si tenéis **Basica** de IBM: **BASICA** y pulsar **ENTER**.

Si tenéis el programa **QuickBasic**: **QB** o **QUICK BASIC** y pulsar **ENTER**.

Si tenéis **TurboBasic**: **TB** y pulsar **ENTER**.

El aspecto del **editor** de cada uno de estos programas puede diferir un poco, no obstante

Fig.24 El procedimiento de control del puerto paralelo depende del sistema operativo utilizado. Mientras que con DOS es suficiente con utilizar instrucciones directas con BASIC, con el sistema operativo Windows hay que utilizar necesariamente un archivo DLL.



las instrucciones a escribir y su proceso es exactamente igual en todos.

Antes de exponer las instrucciones necesarias para encender **todos los diodos LED del Grupo 0**, recordamos una vez más que el puerto paralelo **LPT1** corresponde a la dirección **888 (decimal)**.

NOTA: La palabra **REM** o el signo ' (apóstrofe) son instrucciones BASIC utilizadas para poder escribir **comentarios** en el programa. Estos comentarios son anotaciones aclarativas que **no ejecutan código**.

CLS : REM Borrado de la pantalla

OUT 888,255

REM OUT es la instrucción que activa las salidas

REM 888 es la dirección del Puerto LPT1 (Datos)

REM 255 es el valor decimal para tener un **REM** valor binario de 8 bits con todos a 1

Buffer = INP(889)

REM Buffer es la variable utilizada como memoria

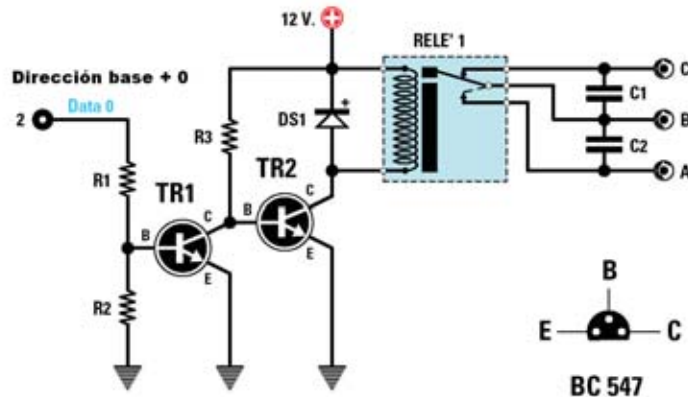
REM en el variable **Buffer** encontraremos

REM un valor decimal equivalente

REM a la combinación binaria leída del puerto

REM INP(889) instrucción que lee la

REM información del puerto LPT1 (Estado)



LISTA DE COMPONENTES

- R1 = 47.000 ohmios 1/4 vatio
- R2 = 100.000 ohmios 1/4 vatio
- R3 = 4.700 ohmios 1/4 vatio
- C1 = 100.000 pF/630 voltios poliéster
- C2 = 100.000 pF/630 voltios poliéster
- DS1 = Diodo 1N.4007
- TR1 = Transistor NPN BC.547
- TR2 = Transistor NPN BC.547

Fig.25 Esquema eléctrico base para realizar un sistema de control por ordenador. También se muestran las conexiones del transistor NPN BC.547.

LISTA DE COMPONENTES

R1 = 680 ohmios 1/4 vatio
R2 = 100 ohmios 1/2 vatio
R3 = 1.000 ohmios 1/2 vatio
C1 = 47.000 pF/400 voltios poliéster
C2 = 47.000 pF/400 voltios poliéster
C3 = 47.000 pF/400 voltios poliéster
Z1 = Impedancia antiparasitaria
DL1 = Diodo LED

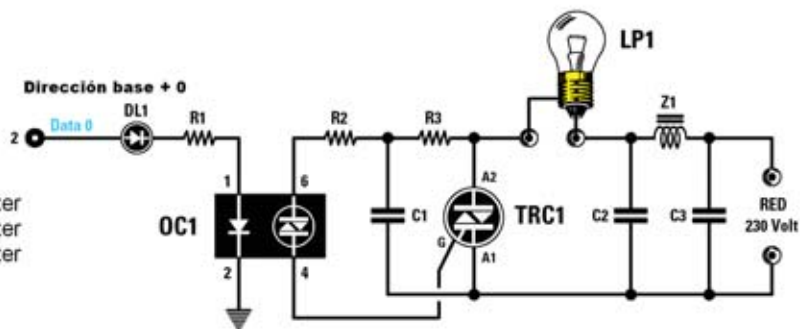
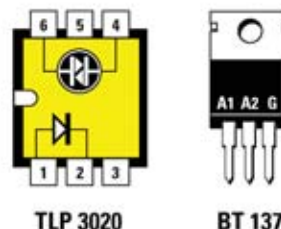


Fig.26 Esquema eléctrico de un circuito que controla cargas a 230 voltios de tipo resistivo desde el puerto paralelo del ordenador. Este esquema utiliza un optoacoplador TLP.3020, cuyas conexiones se muestran junto a las del TRIAC BT.137.



REM 889 es la dirección de LPT1 (Estado)

PRINT BUFFER : REM muestra en pantalla el valor

REM decimal equivalente al valor binario
REM leído en el puerto

Si no hay nada conectado en el puerto el dato leído no significa nada. En el caso de que haya una impresora, el dato leído corresponde al último valor utilizado.

OUT 890,255

REM OUT es la instrucción que activa las salidas

REM 890 es la dirección del Puerto LPT1 (Control)

REM 255 es el valor decimal para tener un
REM valor binario de 8 bits con todos a 1

Si queréis activar todos los bits del puerto de datos con todas las posibles combinaciones en **secuencia** solo hay que utilizar un bucle **FOR-NEXT**:

CLS : REM borrado de la pantalla

FOR i =1 TO 255 : REM bucle desde 1 hasta 255

OUT 888,i

REM activación de las salidas con valor decimal i

NEXT i : REM fin del bucle

ENTORNO WINDOWS

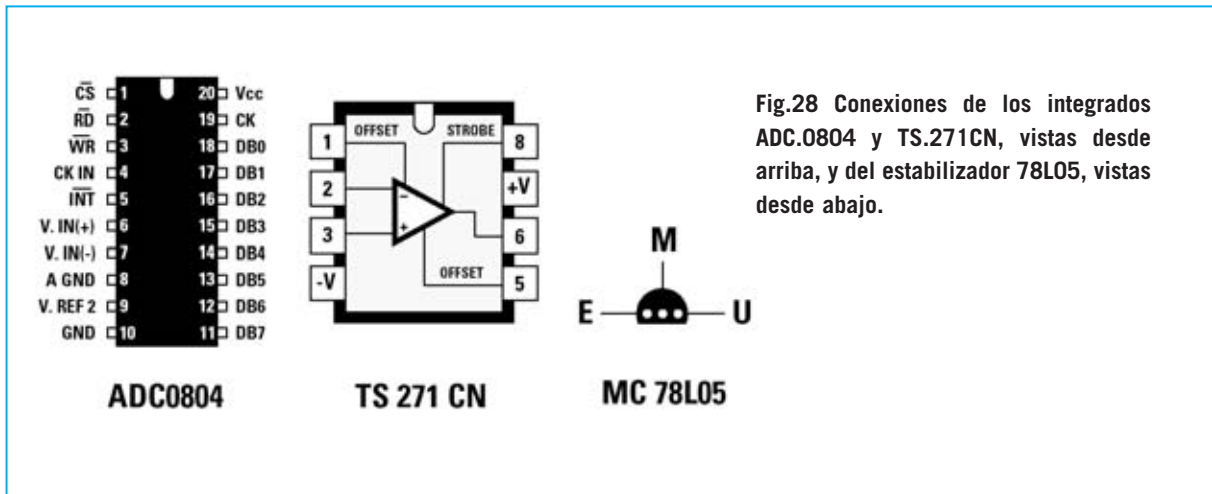
Con el sistema operativo Windows **no** basta con tener un programa como **Visual Basic** y escribir las mismas instrucciones que hemos usado con el intérprete BASIC para DOS, ya que el puerto y el periférico de impresión **no** dialogan directamente con la BIOS del ordenador sino a través de un **programa de gestión** incluido en Windows (ver Fig.24).

Este programa permite que se puedan utilizar los dispositivos, instalando su **driver**, desde cualquier programa diseñado para Windows. De esta forma instalando un **único** driver se puede utilizar el dispositivo desde cualquier programa. En el caso de una impresora, permite que se pueda imprimir desde cualquier programa instalando una sola vez el driver de la impresora.

NOTA: Los **drivers** son los programas de control de un dispositivo y suelen venir incluidos con los dispositivos, en este caso con las impresoras. El driver se utiliza **solo** por el dispositivo para el que ha sido diseñado.

Para “evadir” el control de este programa y acceder al **puerto paralelo** es necesario utilizar un archivo **DLL** (Dynamic Link Library).

Los archivos **DLL** son un conjunto de programas escritos en lenguajes de bajo nivel



(C o Assembler) que contienen un conjunto de rutinas, funciones, iconos, etc. (biblioteca) que pueden ser utilizadas desde otros programas (enlace) y que se utilizan por las aplicaciones software en fase de ejecución (dinámico)

En nuestro caso el archivo DLL está escrito en C y se puede descargar libremente el código fuente desde la Web de Microsoft. Nosotros lo proporcionamos listo para utilizar y probado con Windows 98 SE y Windows XP.

¿Cómo se carga un DLL en un programa?

Para cargar un DLL la manera más simple es crear una carpeta en el ordenador en el que se van a escribir los programas. En esta carpeta hay que copiar, de la carpeta SORGENTILPTTester del CDROM CDR.1588, estos dos archivos:

- El archivo DLL, es decir io.DLL.
- El módulo gestione.bas.

A continuación describimos las operaciones a realizar.

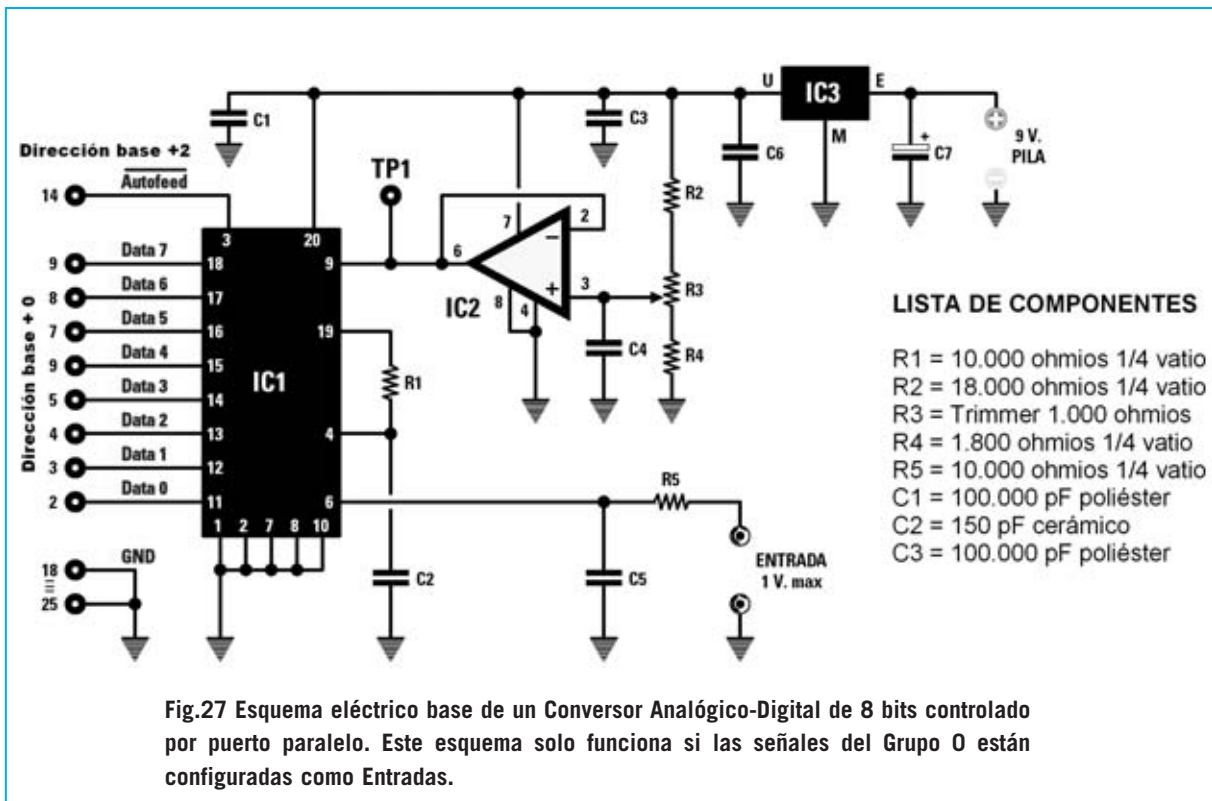


Fig.29. Para ejecutar el programa de conversión A/D hay que hacer click en el botón INICIO. A continuación hay que llevar el cursor sobre PROGRAMAS y localizar la entrada TEST ADC0804. Por último hay que hacer click sobre TEST ADC0804.

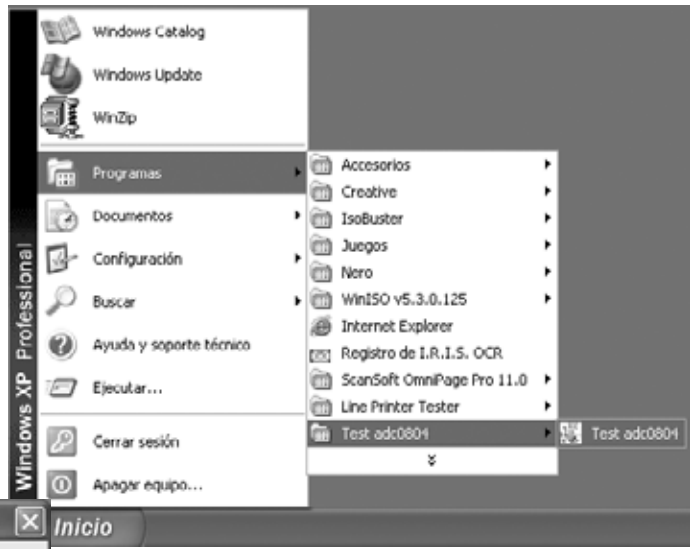


Fig.30 En la ventana principal se puede modificar el tiempo de lectura de la conversión A/D y ver el valor de la conversión (decimal).

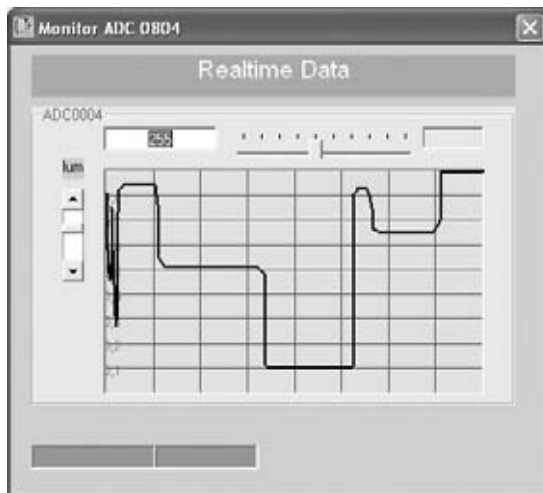


Fig.31 En la ventana REALTIME DATA los valores de tensión que llegan del Conversor A/D se muestran como en un osciloscopio.

En primer lugar hay que abrir **Visual Basic**. A continuación hay que seleccionar la función **Insertar módulo** en el menú **Proyecto**. Automáticamente el ordenador muestra los módulos disponibles, entre los que se encuentra el archivo **gestione.bas**. Hay que seleccionarlo.

El módulo que os proporcionamos contiene la gestión de la **sintaxis** necesaria para escribir las **instrucciones** de acceso al puerto paralelo.

Cuando escribáis las instrucciones para encender **todos** los **diodos LED** del puerto de **Datos** de **LPT1**, de forma similar a como se mostró en entorno DOS, hay que escribir la siguiente instrucción:

PortOut 888, 255

Como se puede observar se utiliza **PortOut** en lugar de la instrucción **Out**.

Para **leer** el puerto de **Estado** de **LPT1** tenemos una instrucción similar a la utilizada en el BASIC bajo DOS, en este caso:

Buffer = PortIn(889)

Es decir, en lugar de **INP(889)** hay que utilizar **PortIn(889)**.

El resto de instrucciones se utilizan igual que en entorno DOS, por lo que estamos en disposición de controlar el **puerto paralelo** en entorno **Windows**.

EJEMPLOS de APLICACIONES

Para complementar los procesos de programación presentados en los epígrafes anteriores exponemos a continuación algunos **esquemas eléctricos** que pueden ser utilizados como **base** para el desarrollo de circuitos más complejos.

ACTIVACIÓN de un RELÉ

Para realizar un sistema automático que pueda **controlar por ordenador la apertura y cierre de contactos** se puede utilizar un relé conectado al puerto paralelo (ver Fig.25). Para excitar el relé con la señal del puerto paralelo (**Data 0**) basta con utilizar dos transistores en configuración darlington.

ACTIVACIÓN de un TRIAC

Quien desee controlar directamente **cargas de 230 V AC** puede utilizar el circuito base mostrado en la Fig.26, circuito que también se controla a través de una señal del puerto paralelo (**Data 0**).

Puesto que hay que manejar la tensión de red, hemos añadido un **optoacoplador (TLP.3020)** que proporciona un **aislamiento** eléctrico máximo entre la baja tensión y la tensión de red (ver **OC1** en la Fig.26). Dentro de este optoacoplador hay un **diodo LED** y un **fotodiodo** que controla la Puerta (**Gate**) del **TRIAC**.

Además hemos añadido un **filtro antiparasitario**, formado por la impedancia **Z1** y por los condensadores **C2-C3**.

Este esquema solo se puede utilizar para cargas de **tipo resistivo**, es decir lámparas incandescentes, calefactores, etc. con potencias no superiores a **500 vatios**. **No** puede utilizarse para controlar **cargas inductivas** como motores o lámparas de neón.

Las **instrucciones** necesarias para activar la salida **Data 0 (pin 2)** en las dos aplicaciones anteriormente expuestas son las siguientes:

DOS / Basic
Out 888,1

Windows / VBasic / DLL
PortOut 888, 1

CONVERSION A/D de 8 bits

Con esta aplicación cerramos la serie de ejemplos para la utilización del puerto paralelo. Como se puede ver en el esquema eléctrico de la Fig.27 hemos utilizado un **Convertor Analógico/Digital de 8 bits** tipo **ADC.0804**.

Cualquier Convertor Analógico-Digital necesita una **tensión de referencia** para poder realizar las conversiones. En nuestro esquema esta tensión está proporcionada por el integrado IC2 que desarrolla la función de **regulador de tensión de precisión**.

Este integrado proporciona la **mitad** de la tensión máxima aceptada en la entrada a través del **divisor** formado por las resistencias **R2-R4** y del trimmer de ajuste **R3**.

La señal a convertir en digital se aplica, a través de la resistencia **R5**, a la entrada de **IC1** (pin 6).

El condensador **C5** se utiliza para eliminar eventuales impulsos que pudieran falsear la lectura, mientras la resistencia **R1** y el condensador **C2** determinan la frecuencia de **reloj** con la que **IC1** realiza las conversiones de analógico a digital.

Cuando la señal **Autofeed**, utilizada como señal de **inicio** de la conversión (conectada al terminal **3** de **IC1**), pasa de estado bajo a alto, el valor digital de **8 bits** correspondiente al valor analógico de entrada se encuentra en los terminales **11-18**.

Para poder **leer** el valor hay que poner el **Grupo 0 (Datos)** en modo **bidireccional**. Esta operación se realiza poniendo a **1** el **bit 5** del **Grupo 2** (ver Fig.6).

Se trata de un **bit de control** no accesible desde el hardware, es decir no encontraremos cables externos que lo conecten a nuestro **Téster para puerto paralelo LX.1588**.

Resumiendo, para configurar el **Grupo 0** del puerto como **entrada** en lugar de como salida hay que añadir el valor **32** decimal (**10000** binario) al valor anterior.

Todo el circuito se alimenta a **5 voltios**.

Esta tensión puede obtenerse directamente de la **fuerza de alimentación** del **ordenador** o de una pila de **9 voltios** y un estabilizador **78L05**.

También se puede utilizar un **alimentador estabilizado**, como el **LX.1486** presentado en la revista **N.203**. Este alimentador se instala en el ordenador como una tarjeta de expansión.

Para ajustar la tensión de referencia hay que controlar un téster, ajustado para medir **tensión DC**, que la tensión en el terminal **9** de **IC1** sea de **0,5 voltios**. El valor de la tensión se **ajusta** con el **trimmer R3**.

SOFTWARE para el CONVERTOR A/D

Llegado este punto podéis escribir las instrucciones para comprobar que todo funciona o bien utilizar el programa **ADC0804** (incluido en el CDROM **CDR.1588**). Para instalar este programa hay que seguir las indicaciones anteriormente descritas para el programa **LPT Tester** (ver Figs.13-20) escribiendo **D:\ADC0804\setup.exe** en lugar de **D:\LPTTester\setup.exe** (ver Fig.14).

NOTA: **D:** corresponde a la letra asignada a la unidad donde se ha introducido el CDROM. Si se corresponde a otra letra, **D:** ha de sustituirse por la letra correspondiente a la unidad.

EL PROGRAMA ADC0804

Para abrir el programa de conversión A/D hay que comenzar haciendo click en el botón **Inicio** del Escritorio de Windows. A continuación hay que llevar el cursor sobre **Programas** y, en la lista que se despliega, hay que localizar la entrada **Test adc0804** y posicionar el cursor sobre ella. Por fin, en el último cuadro desplegado, hay que hacer click sobre **Test adc0804**. La secuencia de estas operaciones se puede ver en la Fig.29.

La ventana principal incluye **tres recuadros de ajuste**. En el recuadro superior se puede ajustar (en **segundos**) el **tiempo** de lectura y conversión del ADC. El valor predeterminado es **0,5 segundos**. Si en este recuadro se

escribe, por ejemplo, el valor **0,05**, el conversor AD realiza una lectura cada **5 centésimas de segundo**.

Los dos recuadros de la parte inferior muestran, respectivamente, el **valor** decimal correspondiente a la combinación binaria del **Grupo 0 (Datos)** del puerto paralelo y el valor de la **tensión**.

Haciendo click en el botón **Monitor** se abre la ventana **Realtime Data** (ver Fig.31), aquí se muestra un pequeño osciloscopio virtual. En esta ventana se representan de forma gráfica los valores leídos, como en un **osciloscopio**.

Personalizar el programa del Conversor A/D

El valor en **reposo** de la señal **#Autofeed**, conectada al terminal **3** del integrado **IC1**, es el nivel lógico **1**. Cuando pasa de nivel lógico **1** a nivel lógico **0** se inicia la **conversión**. Cuando vuelve a nivel lógico **1** el valor de la señal analógica está disponible, en formato

```
Private Sub Timer1_Timer()
' CS bajo e IRQ5 bidireccional
' 10 1011 = H2b = 43
PortOut porta2, &H2B
' subrutina de retardo
Make_Delay (Delay_Time)
' CS alto e IRQ5 bidireccional
' 10 1001 = H29 = 41
PortOut porta2, &H29
' subrutina de retardo
Make_Delay (Delay Time)
' lectura de datos del Puerto
datiport0 = PortIn(porta0)
' muestra el dato AD en la ventana Text
Text2.Text = datiport0
' calculo de volt con 2 decimales -----
' transformar el valor AD en volt
' multiplicar por 100 para tener 4 numeros
' antes de la coma
uno = (datiport0 / 255) * 100
' coger solo la parte entera del numero
perlett = Int(unos)
' dividir por 100 para obtener volt
' real con solo 2 decimales
VOLT(1) = perlett / 100
' muestra volt en la ventana Text
Text3.Text = VOLT(1)
'----- gestion osciloscopio virtual -----
grafM.Label14.Caption = VOLT(1)
grafM.Text3.Text = datiport0
per10 = perlett
grafM.grafdynamic1
End Sub
```

binario de 8 bits, en los **8 cables del Grupo 0** del puerto paralelo.

En la imagen adjunta se puede observar la rutina Visual Basic que administra la adquisición de los datos. Se trata de la rutina que, como se puede ver en el fuente incluido en el CDROM **CDR.1588**, es administrada por un **temporizador (timer)** que lee los datos a intervalos regulares de tiempo.

Para tener un **nivel lógico 0** en el terminal **14** del **Grupo 2** hay que poner el bit a **1**, ya que esta señal está **invertida** (ver **#Autofeed** en la Fig.6).

Ahora se puede probar con el programa **LPT Tester**: Para tener todas las salidas del **Grupo 2** a **0** con el bit **IRQ5** a **1**, transformando el **Grupo 0** en bidireccional configurándolo como **entrada**, hay que tener la siguiente combinación binaria (**#Autofeed** a nivel **bajo** y **entrada** activada):

nc	nc	IRQ5	IRQ4	17	16	14	13
0	0	1	0	1	0	1	1
		32		8		2	1

La suma de los pesos de los bits a **1** da como resultado:

$$32 + 8 + 2 + 1 = 43$$

que en hexadecimal es: **2B**. Luego la instrucción es:

PortOut porta2, &H2B

Para tener **#Autofeed** a nivel **alto** y la **entrada** activada:

nc	nc	IRQ5	IRQ4	17	16	14	13
0	0	1	0	1	0	0	1
		32		8			1

La suma de los pesos de los bits a **1** da como resultado:

$$32 + 8 + 1 = 41$$

que en hexadecimal es: **29**. Luego la instrucción es:

PortOut porta2, &H2B

NOTA: En las instrucciones hemos utilizado valores **hexadecimales** aunque también pueden utilizarse valores **decimales**.

Las instrucciones a utilizar, tanto en entorno **DOS** como en entorno **Windows**, son las siguientes:

DOS / BASIC
DLL

Out 890,41
Out 890,43 'bit 5 a 1
datiport0 = Inp (888)
(888)

Windows / VBasic /

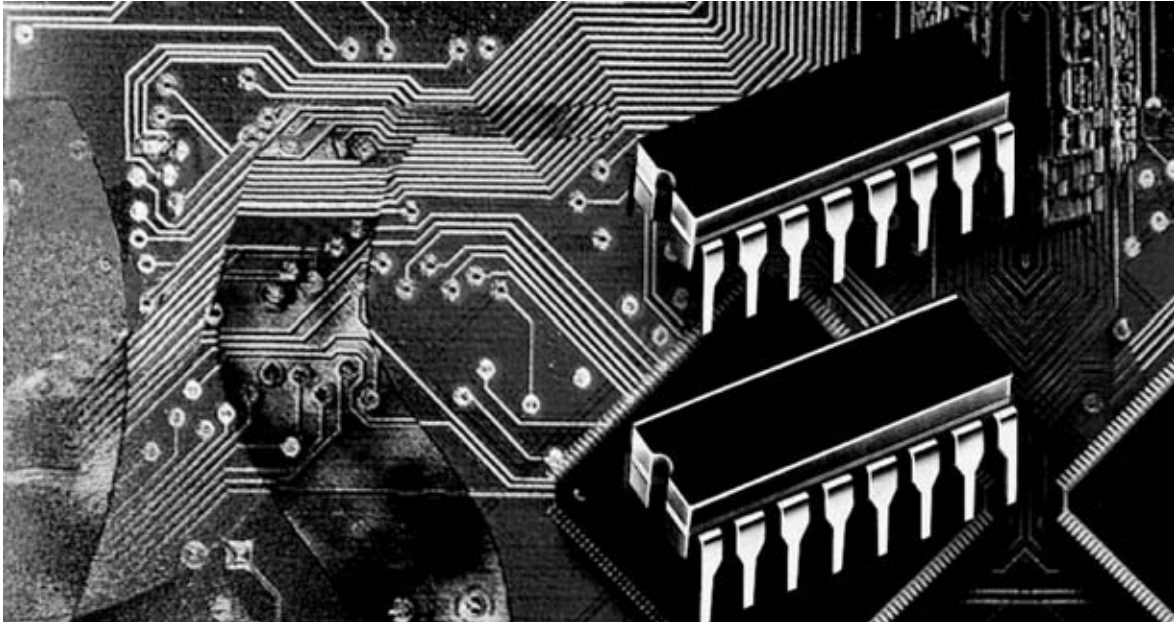
PortOut 890,41
PortOut 890,43
datiport0 = PortIn
(888)

PRECIO de REALIZACIÓN

LX.1588: Precio de todos los componentes necesarios para realizar el **Téster para Puerto Paralelo** (ver Figs.9-11), incluyendo circuito impreso, contenedor de plástico con pletina autoadhesiva de aluminio perforada y **serigrafiada**, conectores SUB-D 25 macho y hembra, resistencias, diodos LED, **incluido** también el CDROM **CDR.1588****32,40 €**

CC.1588: Circuito impreso**1,95 €**

ESTOS PRECIOS NO INCLUYEN I.V.A.



Programación con microcon

Una de las peculiaridades del lenguaje Assembler para los micros ST7 es la posibilidad de direccionar los operandos de una instrucción con varios modos diferentes. Hoy vamos a aprender a reconocer y a utilizar los modos de direccionamiento indirectos y los modos indexados indirectos.

Algunos lectores, después de haber leído los artículos anteriores sobre direccionamiento, nos han escrito preguntándonos cual es el **modo de direccionamiento** mejor. A esta pregunta sólo podemos contestar que no existe un modo mejor que los otros, ya que de ser así solo existiría ese modo de direccionamiento. Existen **diferentes modos** ya que cada uno es el más adecuado en función de la estructura del programa.

Por esta razón se hace necesario conocer todos los **modos de direccionamiento** y, sobre todo, comprender su **funcionamiento**. Los ejemplos que os proponemos son muy sencillos y están realizados para alcanzar este objetivo.

Cuando nos ocupemos de cada una de las **instrucciones Assembler**, haremos referencia a los modos de direccionamiento ya analizados. Una vez leídos estos artículos ya no tendréis dudas sobre la elección de los modos a utilizar en vuestros programas.

DIRECCIONAMIENTO INDIRECTO

Para explicar la lógica y las características del direccionamiento **indirecto** vamos a hacer un desarrollo similar al del direccionamiento **directo**, presentado en la revista **N.233**.

Recordando el modo de direccionamiento **directo**, el operando define el **valor contenido** en la dirección de memoria, con instrucciones del tipo: **ld a,pippo**

El valor contenido en la dirección de memoria en la que está definida la variable **pippo** se obtiene directamente y se carga en el acumulador **A**. En modo Indirecto una instrucción de este tipo se escribe de la siguiente forma:

ld a,[pippo]

En el acumulador **A** no se carga el valor contenido a la dirección de memoria **pippo** sino que, en este caso, se carga en el acumulador el **contenido** de la **posición de memoria** cuya dirección se encuentra en la dirección de memoria **definida** por la variable **pippo**. Por esta razón se llama direccionamiento indirecto: La variable **no** contiene el **operando** sino la **dirección** que contiene el **operando**.

en el variable **pippo** sino el valor contenido en la **dirección** a la que **apunta** el contenido del variable **pippo**, como se muestra gráficamente en la Fig.2.

Después de la ejecución de la instrucción el registro **A** contiene el valor **18h** (ver Fig.2). La instrucción con modo **directo**, es decir con el operando **sin corchetes**, habría cargado en **A** directamente el valor **9Ch**.

La instrucción **ld a,[pippo]** en formato ejecutable tiene un **código de operación (op-code)** **92B67B**, donde **92** indica que se trata de una instrucción de direccionamiento **indirecto**, **B6** es el código propio de la instrucción **ld a,[dirección]** y **7B** es la dirección de la variable con etiqueta **pippo**.

troladores ST7 LITE 09 (5)

En principio puede parecer un poco difícil de entender, vamos a aclararlo más. En la Fig.1 hemos detallado las partes que componen una instrucción Assembler en código fuente.

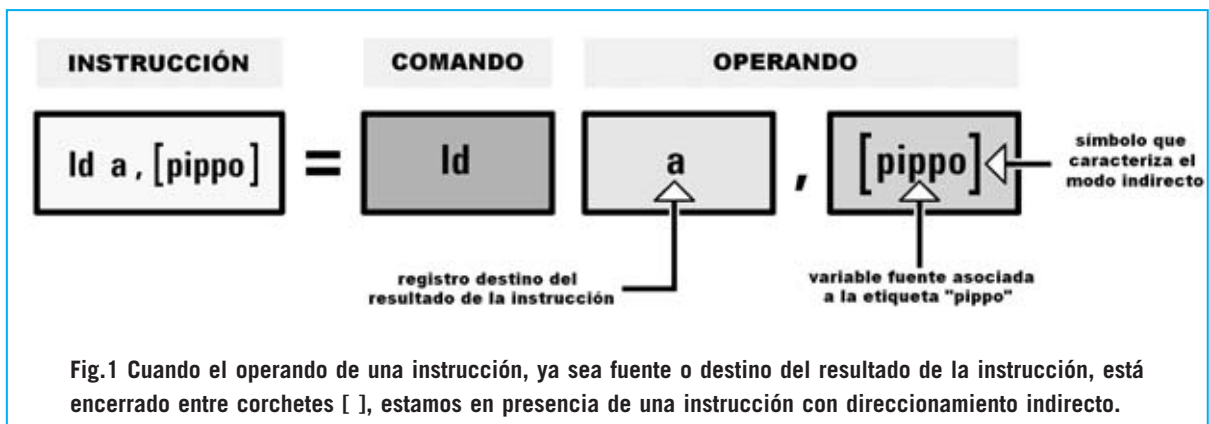
Supongamos que la variable **pippo** ha sido definida en la dirección **007Bh** de memoria RAM. La instrucción:

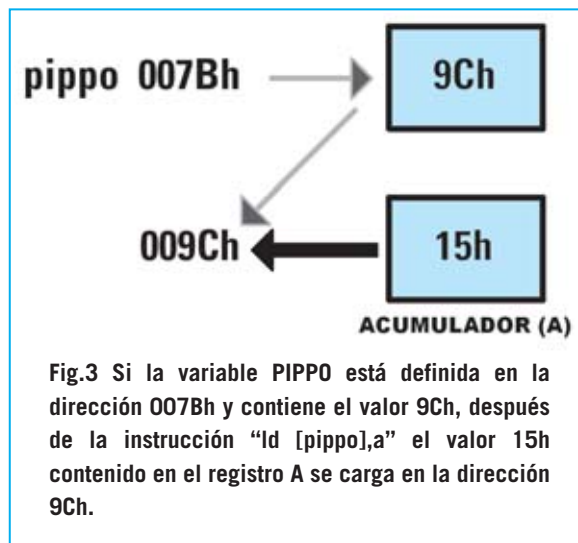
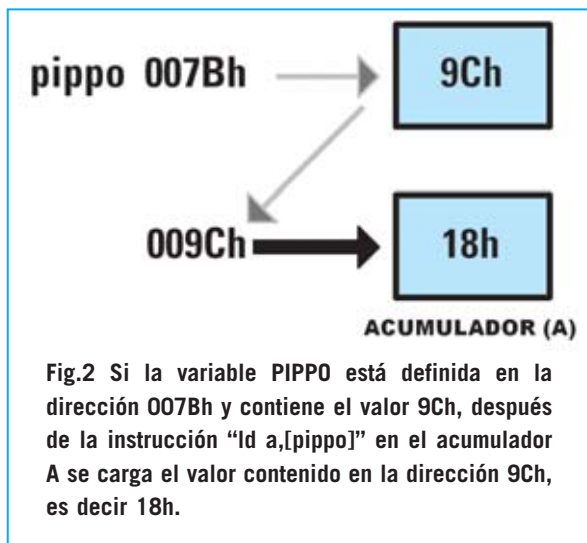
ld a,[pippo]

trabaja con direccionamiento **indirecto**. En el acumulador **A** **no** se carga el valor contenido

Dado que el código de operación tiene **tres bytes (92B67B)**, la dirección siguiente del Contador de Programa (Program Counter) corresponde a la dirección de la instrucción + **3 bytes**. Por tanto, si la instrucción estuviera en **043Dh**, el Contador de Programa pasaría a **043Dh + 3 = 0440h**.

La primera característica a tener presente en el direccionamiento **indirecto** es que también permite acceder de forma **indirecta** a la dirección de memoria del **operando destino**.





En otras palabras, también existe la instrucción:

ld [pippo],a

Obviamente en este caso el contenido del registro **A** no se carga en la dirección en la que está definida la variable **pippo** sino en la dirección a la que **apunta** el contenido de la variable **pippo**.

Suponiendo que el registro **A** contiene el valor **15h** y **pippo** está definida, como ya hemos mencionado, en la dirección **007Bh** (que contiene el valor **9Ch**), esta instrucción carga el valor **15h** en la dirección **9Ch** (ver Fig.3).

La instrucción con modo **directo**, es decir con el operando **sin corchetes**, habría cargado directamente en **pippo** (dirección **007Bh**) el valor **15h**.

También la instrucción **ld [pippo],a** tiene un código de operación de tres bytes, en este caso **92B77B**, donde **92** indica que se trata de una instrucción de direccionamiento **indirecto**, **B7** es el código propio de la instrucción **ld [dirección],a** y **7B** es la dirección de la variable con etiqueta **pippo**.

Ya que el código de operación tiene **tres bytes (92B77B)**, la dirección siguiente del Contador de Programa corresponde a la dirección de la instrucción + **3 bytes**. Por tanto, si la instrucción

estuviera en **043Dh**, el Contador de Programa pasaría a **043Dh + 3 = 0440h**.

Hay que señalar llegado este punto que, al igual que el modo directo, también el modo de direccionamiento indirecto puede **indexarse**, utilizando uno de los registros índice **X -Y**, para acceder **indirectamente** a una **posición de memoria**.

Por tanto, el **direccionamiento indirecto** tiene cuatro diferentes modos operativos que a continuación pasamos a analizar pormenorizadamente:

- Indirecto Corto (Short)
- Indexado Indirecto Corto (Short)
- Indirecto Largo (Long)
- Indexado Indirecto Largo (Long)

INDIRECTO CORTO (SHORT)

Una vez aclarado el **concepto** de direccionamiento **indirecto**, vamos a presentar un ejemplo que ayude a entender mejor esta forma de direccionamiento.

En primer lugar definimos en **Data Memory** una serie de variables:

(0087h)	VALAD1	DS.B 1
.....
(008Bh)	VALAD2	DS.B 1
.....
(0094h)	PUNTAT	DS.B 1

NOTA: Los valores **hexadecimales** entre paréntesis situados a la izquierda representan **direcciones de memoria**. Se trata de valores **hipotéticos** expuestos con fines didácticos. Las tres direcciones son intencionalmente no contiguas.

Las siguientes instrucciones se alojan en **Program Memory**:

(FA13h)	ld	a,#VALAD2
(FA15h)	ld	PUNTAT,a
(FA17h)	call	caricad
.....
(FAD0h)	ld	a,#VALAD1
(FAD2h)	ld	PUNTAT,a
(FAD4h)	call	caricad
.....
(FAE2h)	caricad	
.....
(FAE9h)	btjf	ADCCSR,#7,\$
(FAFCh)	ld	a,ADCDR
(FAFFh)	ld	[PUNTAT],a
(FB03h)	ret	

Empezamos analizando el programa con la primera instrucción:

(FA13h) **ld** **a,#VALAD2**

Esta instrucción carga en el acumulador **A** la dirección de la variable **VALAD2**. En efecto, el símbolo corchete “#” que precede al operando indica que se trata de una instrucción con direccionamiento **inmediato** (ver revista **N.233**).

Al compilar esta instrucción se traduce a formato ejecutable con un código de operación **A68B**, donde **A6** es la traducción en formato ejecutable de **ld a,#valor**, mientras que **8B** es el valor **inmediato** del operando (que corresponde a la dirección de memoria de la variable **VALAD2**)

Cuando se ejecuta la instrucción el micro carga el valor **8Bh** en el acumulador **A**.

Con la instrucción:

(FA15h) **ld** **PUNTAT,a**

de direccionamiento directo, se carga en la variable **PUNTAT** el valor contenido en el acumulador **A**.

Al compilar esta instrucción se traduce a formato ejecutable con un código de operación **B794**, donde **B7** es la traducción en formato ejecutable de **ld variable,a**, mientras que **94** es la dirección de memoria de la variable **PUNTAT**.

Cuando se ejecuta esta instrucción el micro carga en la variable **PUNTAT** el contenido del acumulador **A**, es decir **8Bh** (que corresponde a la dirección de memoria de la variable **VALAD2**).

La instrucción siguiente, es decir:

(FA17h) **call** **caricad**

lanza la ejecución de la subrutina **caricad**. Por tanto el programa salta a la etiqueta **caricad**:

(FAE2h) **caricad**

y almacena en la **pila (Stack Memory)** la dirección de retorno.

NOTA: El funcionamiento de la pila y de las subrutinas ha sido explicado en artículos anteriores.

La instrucción de la subrutina:

(FAE9h) **btjf** **ADCCSR,#7,\$**

realiza una lectura A/D y almacena el resultado en una variable.

El funcionamiento del **Convertor A/D** se tratará en próximos artículos. De momento adelantamos que el registro **ADCCSR** controla la conversión A/D y que la instrucción **btjf ADCCSR,#7,\$** literalmente significa “salta a \$ si el bit 7 del registro de control **ADCCSR** es 0”.

NOTA: la instrucción **btjf** es el acrónimo de **Bit Test Jump if False**, es decir controla el bit y salta si 0.

El terminal **7** del registro **ADCCSR** se pone a **0** cuando se activa una conversión **A/D**

(Analógico/Digital) y queda así hasta que se realiza la conversión. Una vez realizada la conversión este terminal pasa a **1** y el dato digital convertido se almacena en el registro **ADCDR**.

El símbolo **\$** representa al “Contador de Programa actual”, que en nuestro ejemplo vale **FAE9h**. Por tanto, la instrucción **btjf ADCCSR,#7,\$** realiza saltos sobre sí misma hasta que se realiza la conversión A/D.

Suponiendo que el resultado de la conversión A/D es **1Fh**, el registro **ADCDR** contendrá este valor.

Ahora, como hemos expuesto, vamos a almacenar este resultado en una variable, comenzando por moverlo al acumulador **A** con la siguiente instrucción:

```
(FAFCh)      Id      a,ADCDR
```

y, a continuación:

```
(FAFFh)      Id      [PUNTAT],a
```

A primera vista puede parecer que el valor que se encuentra en el acumulador **A** se almacena en el variable **PUNTAT**, no es así ya que los corchetes indican la utilización de un **direccionamiento indirecto corto**.

Al compilar esta instrucción se traduce a formato ejecutable con un código de operación de tres bytes: **92B794**, donde **92** indica que se trata de una instrucción de direccionamiento indirecto, **B7** es la traducción en formato ejecutable de **ld variable,a**, mientras que **94** es la dirección de memoria de la variable **PUNTAT**.

Cuando se ejecuta la instrucción, el valor presente en el acumulador **A**, es decir **1Fh**, se carga en la dirección de memoria indicada por el valor presente en la variable **PUNTAT**.

Puesto que en **PUNTAT** cargamos la dirección de la variable **VALAD2**, es decir **8Bh**, el valor **1Fh** se almacena en **VALAD2** (dirección **8Bh**). Con este modo de direccionamiento la variable **PUNTAT** se ha utilizado como **puntero de memoria**.

Con la instrucción:

```
(FB03h)      ret
```

finaliza la subrutina provocando el **retorno** al programa principal.

El programa continúa con las siguientes instrucciones:

```
(FAD0h)      Id      a,#VALAD1
(FAD2h)      Id      PUNTAT,a
(FAD4h)      call     caricad
```

Llegados a este punto el valor resultante de la conversión A/D se almacena, mediante estas instrucciones, en la variable **VALAD1**.

Con este ejemplo hemos querido mostrar como utilizando el modo de direccionamiento **indirecto** se pueden utilizar subrutinas con parámetros sin modificarla. En efecto, solo hemos modificado la dirección de la variable, mientras que la subrutina **caricad** ha quedado inalterada.

El modo **indirecto** se define como **short (corto)** cuando el direccionamiento se realiza con una **variable** de **un byte** de longitud, lo que implica un rango de **00h** a **FFh**.

INDEXADO INDIRECTO CORTO (SHORT)

El modo **indexado indirecto corto** es un modo de direccionamiento análogo al anterior pero que utiliza el registro índice **X** o el **Y**, de ahí el término “**indexado**”.

Merece la pena recordar que, como describimos en la revista **N.233**, el modo **indexado** funciona sumando la **dirección** de la **variable**, denominada también **desplazamiento (offset)**, al valor contenido en el registro índice **X** o **Y**.

El ejemplo utilizado en aquella ocasión fue **loop clr (PROVS,x)**.

Como **PROVS** fue definida en la dirección **8Bh** y el registro **X** tenía el valor **9h**, la instrucción **clr** borra el byte contenido en la dirección **8Bh + 9h** y es decir en **94h**.

En el ejemplo que presentamos a continuación hemos realizado un programa donde en puntos diferentes se lanzan **subrutinas** de **conversión A/D**, cuyos resultados son almacenados en **dos variables** diferentes y los tres bytes siguientes a cada una.

Por tanto, en primer lugar **definimos** las **variables** en **Data Memory**, como en el ejemplo anterior, pero con una pequeña modificación.

Primero definimos un área de **4 bytes** asociando la dirección del primer byte a la variable **VALAD1**, a continuación definimos un área de **4 bytes** asociando la dirección del primer byte a la variable **VALAD2**. Simplificando podemos decir que **VALAD1** y **VALAD2** son **variables de 4 bytes**.

(0087h)	VALAD1	DS.B 4
(008Bh)	VALAD2	DS.B 4
.....
(0094h)	PUNTAT	DS.B 1

Ahora escribimos las **instrucciones** del programa de ejemplo:

(FA13h)	ld	a,#VALAD2
(FA15h)	ld	PUNTAT,a
(FA17h)	clr	x
(FA18h)	call	caricad
.....
(FAD0h)	ld	a,#VALAD1
(FAD2h)	ld	PUNTAT,a
(FAD4h)	clr	x
(FAD5h)	call	caricad
.....
(FAE2h)	caricad	
.....
(FAE9h)	btjf	ADCCSR,#7,\$
(FAFCh)	ld	a,ADCDR
(FAFFh)	ld	([PUNTAT],x),a
(FB02h)	inc	x
(FB03h)	cp	x,#4
(FB05h)	jrc	caricad
(FB07h)	ret	

El primer grupo de cuatro instrucciones:

(FA13h)	ld	a,#VALAD2
(FA15h)	ld	PUNTAT,a

(FA17h)	clr	x
(FA18h)	call	caricad

es prácticamente igual al del ejemplo anterior. La única diferencia es la inserción de la instrucción:

(FA17h)	clr	x
---------	------------	----------

con la que se **borra (clr)** el registro **X**.

La situación antes de la ejecución de la subrutina **caricad** es la siguiente:

- El registro **X** contiene **0**.
- La variable **PUNTAT** contiene el valor **8Bh**, que es la dirección de inicio de **VALAD2**.

La instrucción:

(FA18h)	call	caricad
---------	-------------	----------------

lanza la subrutina:

(FAE2h)	caricad	
.....
(FAE9h)	btjf	ADCCSR,#7,\$
(FAFCh)	ld	a,ADCDR
(FAFFh)	ld	([PUNTAT],x),a
(FB02h)	inc	x
(FB03h)	cp	x,#4
(FB05h)	jrc	caricad
(FB07h)	ret	

Las instrucciones de la subrutina son las mismas que las utilizadas en el ejemplo anterior, hasta llegar a la instrucción que utiliza el modo **indexado indirecto corto**:

(FAFFh)	ld	([PUNTAT],x),a
---------	-----------	-----------------------

Como en el ejemplo anterior, el acumulador **A** contiene el valor de la conversión A/D, es decir **1Fh**. Al trabajar en modo **indexado** la dirección de la variable constituye el desplazamiento al que hay que añadir el valor contenido en el registro **X**.

El registro índice **X** contiene el valor **0** y la variable **PUNTAT** está encerrada entre corchetes [], por lo tanto no se utiliza su dirección de su definición (**94h**) sino el valor

contenido en esta dirección, es decir **8Bh**, que es la dirección de **VALAD2**.

El resultado de la ejecución de esta instrucción es que, la **primera vez** que se ejecuta la subrutina, el valor **1Fh** contenido en el acumulador **A** se almacena en la dirección de memoria **8Bh + 0** (es decir **VALAD2+0**).

Hemos indicado “primera vez” ya que, como se muestra en el grupo de instrucciones que exponemos a continuación, la **subrutina** se ejecuta **cuatro veces** antes de volver al programa principal.

(FB02h)	inc	x
(FB03h)	cp	x,#4
(FB05h)	jrc	caricad
(FB07h)	ret	

En efecto, con la instrucción **inc x** se incrementa en **1** el valor del registro **X**, por lo tanto la **segunda vez** no contiene el valor **0** sino **1**.

Con la instrucción **cp x,#4**, el valor contenido en el registro **X** se compara con el valor inmediato **4**, caracterizado por el símbolo **#**. Si el valor es **menor** que **4** el **flag de Acarreo (Carry)** se pone a **1**, si es **mayor o igual a 4** se pone a **0**.

Con la instrucción siguiente (**jrc caricad**) el programa salta a la subrutina **caricad** si el **flag de Acarreo** vale **1** (**jump relative carry**), si vale **0** se ejecuta la instrucción siguiente, que es la instrucción de **retorno** de subrutina (**ret**).

Ya que en este momento el valor contenido en el registro **X** es **1**, es decir menor que **4**, el **flag de Acarreo** vale **1** y, por lo tanto, el programa ejecuta de nuevo la parte de la subrutina relativa a la conversión A/D, hasta llegar a la instrucción:

(FAFFh)	ld	([PUNTAT],x),a
---------	-----------	-----------------------

Esta vez el nuevo valor convertido, disponible en el acumulador **A**, se almacena en la dirección **8Bh + 1** o (es decir **VALAD2+1**).

Posteriormente se vuelve a **incrementar** el registro **X**, pasando a valer **2**, y a compararse con **4**. Ya que este valor es **menor** que **4** el **flag de Acarreo** vale **1** y permite de nuevo la ejecución

de la subrutina. En consecuencia, el valor de la conversión se almacena en **8Bh + 2** (**VALAD2+2**).

La siguiente vez se repetirá el proceso almacenándose el valor convertido en **8Bh + 3** (**VALAD2+3**). Ahora bien, cuando el registro **X**, a fuerza de ser incrementado, llega a contener el mismo valor con el que se compara, es decir **4**, lo que sucede después de **4** ciclos de ejecución de la **subrutina**, el **flag de Acarreo** se pone a **0**.

Dado que la condición de salto a la subrutina **caricad (jrc)** no se cumple, el programa continúa con la instrucción **ret**, volviendo así al programa principal, ejecutándose por tanto las restantes instrucciones:

(FAD0h)	ld	a,#VALAD1
(FAD2h)	ld	PUNTAT,a
(FAD4h)	clr	x
(FAD5h)	call	caricad

Resumiendo, el programa va almacenando el resultado de la conversión A/D en las siguientes direcciones:

87h + 0 (**VALAD1+0**)
87h + 1 (**VALAD1+1**)
87h + 2 (**VALAD1+2**)
87h + 3 (**VALAD1+3**)

Utilizando el modo **indexado indirecto corto** hemos podido direccionar áreas diferentes y consecutivas de memoria escribiendo una única subrutina de almacenamiento. En este modo los corchetes que caracterizan el direccionamiento **indirecto** se escriben en los extremos de una **variable** (denominada **desplazamiento**), que puede tener un valor máximo de **FFh** al tener un **byte** de longitud (**corta**).

INDIRECTO LARGO (LONG)

El modo **indirecto** se define como **largo (long)** cuando direcciona áreas de memoria superiores a **FFh** utilizando como operando una **variable de dos bytes** de longitud. En realidad hablar de variable de **dos bytes** no es del todo cierto sino una simplificación, ya que en el lenguaje **Assembler para ST7** no existe la posibilidad de asociar una **longitud** a las variables definidas. La **variable** siempre identifica únicamente la **dirección de memoria** en la que ha sido definida,

y como este lenguaje es de un microcontrolador con tecnología de **8 bits**, las instrucciones tratan valores contenidos en **1 byte**.

Entonces ¿qué quiere decir y cómo funciona el direccionamiento **largo (long)**?

Cuando se habla de una variable de **dos bytes** de longitud, como por ejemplo:

```
(0094h) PUNTAT DS.B 2
```

se **definen 2 bytes (DS.B 2)** y se **asocia** a la **variable (PUNTAT)** la **dirección de memoria (0094h)** del **primer byte** (nótese que **no** se asocian dos bytes a la variable ya que no es posible tal operación).

La directiva **DS.B** quiere decir “Define Space Bytes”, y el número que la sigue indica el **número de bytes** de “espacio” definido o, mejor dicho, **reservados**. Si quisiéramos ahora definir otra variable, la siguiente dirección de memoria sería: **0094h + 2**, es decir **0096h**.

De esta forma reservamos un área de memoria de **2 bytes** a partir de la dirección **0094h**. A esta dirección hemos asociado la etiqueta **PUNTAT**.

Para explicar el direccionamiento **indirecto largo** vamos a utilizar un programa de ejemplo que incluye una subrutina que efectúa una serie de operaciones obteniendo los datos de una tabla de valores constantes. Esta subrutina se llama desde diferentes puntos del programa, utilizando valores contenidos en tablas diferentes.

En primer lugar definimos una **variable** que nos sirve para este tipo de direccionamiento:

```
(0094h) PUNTAT DS.B 2
```

A continuación definimos las **tablas** que contienen los valores: **3 tablas**, cada una con **4 elementos** y con valores predefinidos diferentes.

```
(FC03h) TB_01 DC.B 01h,03h,05h,07h
(FC07h) TB_02 DC.B 02h,04h,06h,08h
(FC0Ch) TB_03 DC.B 10h,20h,30h,40h
```

Para definir las tablas hemos utilizado la directiva **DC.B**. En este caso los valores situados a la derecha de **DC.B** no definen el tamaño

reservado en bytes sino el **valor** contenido en cada byte a partir de la **primera dirección de memoria** asociada a la etiqueta de la tabla.

En otras palabras, considerando la primera tabla, a la dirección de memoria **FC03h** le hemos asociado la etiqueta **TB_01** y hemos insertado en este byte el valor **01h**. Luego, en el byte siguiente, es decir en la dirección **FC04h**, hemos insertado el valor **03h**, y así sucesivamente.

A continuación escribimos las **instrucciones** del programa de ejemplo:

```
(FA13h) Id a,#TB_02.h
(FA15h) Id PUNTAT,a
(FA18h) Id a,#TB_02.l
(FA1Bh) Id {PUNTAT+1},a
(FA1Eh) call CALCOLT
.....
(FA23h) Id a,#TB_01.h
(FA25h) Id PUNTAT,a
(FA28h) Id a,#TB_01.l
(FA2Bh) Id {PUNTAT+1},a
(FA2Eh) call CALCOLT
.....
(FA30h) Id a,#TB_03.h
(FA32h) Id PUNTAT,a
(FA35h) Id a,#TB_03.l
(FA38h) Id {PUNTAT+1},a
(FA3Bh) call CALCOLT
.....
(FB51h) CALCOLT Id a,[PUNTAT.w]
.....
(FBD4h) ret
```

Con la primera instrucción, que utiliza direccionamiento **inmediato** (caracterizado por la presencia del símbolo #):

```
(FA13h) Id a,#TB_02.h
```

cargamos en el acumulador **A** el valor correspondiente a la dirección en la que **TB_02** está definida.

Esta tabla está definida en la dirección **FC07h**, que es de **dos bytes**. Ahora bien, el acumulador **A** solo puede contener un byte, es decir a lo sumo el valor **FFh**. ¿Qué hacer?.

Veamos en detalle el Assembler para ST7.

Seguramente todos habréis apreciado que la después de **TB_02** aparece “. h”. En este caso **no** se hace referencia a notación hexadecimal, sino que indica que se toma en consideración el valor **high** de la dirección de **TB_02**, es decir el **byte más significativo**. Al procesarse la instrucción, en el registro acumulador **A** se carga, por tanto, el valor **FC**.

Con la instrucción siguiente:

```
(FA15h)      Id      PUNTAT,a
```

guardamos el valor **FC** en la variable **PUNTAT**, es decir en la dirección **0094h**.

El programa continúa con la instrucción:

```
(FA18h)      Id      a,#TB_02.I
```

En este caso **TB_02** termina con “.I”, que indica que se toma en consideración el valor **low** de la dirección de **TB_02**, es decir el **byte menos significativo**. Al procesarse la instrucción, en el registro acumulador **A** se carga, por tanto, el valor **07**.

Resulta evidente que de este modo se “divide” la dirección de **TB_02** en **dos bytes** independientes.

Evidentemente antes de utilizar los **dos bytes** tenemos que guardar también este byte, pero siguiendo un **orden** preciso: El byte menos significativo debe guardarse en la dirección siguiente a la que hemos salvado el byte anterior. Esta operación se realiza con la instrucción siguiente:

```
(FA1Bh)      Id      {PUNTAT+1},a
```

Las dos **llaves { }** que encierran el operando hacen referencia al uso de una expresión. En nuestro caso queremos cargar el valor contenido en el acumulador **A** en la dirección **PUNTAT+1**.

El Compilador Assembler procesará esta expresión y generará un ejecutable donde la dirección del operando se convierte en **0095h**, que es precisamente la dirección del byte siguiente a **PUNTAT**.

Seguramente alguien se esté preguntado por qué hemos utilizado este método, ya que

podríamos haber definido inicialmente:

```
(0094h)      PUNTAT1 DS.B 1  
(0095h)      PUNTAT2 DS.B 1
```

y así, en vez de escribir:

```
(FA1Bh)      Id      {PUNTAT+1},a
```

podríamos haber escrito:

```
(FA1Bh)      Id      PUNTAT2,a
```

obteniendo el mismo resultado. Nosotros creemos más conveniente el método de la “**variable de 2 bytes**” ya que así los datos almacenados siempre son consecutivos. Utilizando “**dos variables de 1 byte**” se corren más riesgos al poder cometer algún error con más facilidad al acceder a los datos y la interpretación de lo que realiza el programa es algo más compleja.

Después de esta aclaración volvemos a nuestro ejemplo, donde encontramos la instrucción siguiente:

```
(FA1Eh)      call     CALCOLT
```

que llama a la subrutina:

```
(FB51h)      CALCOLT Id      a,[PUNTAT.w]  
.....  
.....  
(FBD4h)      ret
```

En esta subrutina la primera instrucción utiliza direccionamiento **indirecto largo**:

```
(FB51h)      CALCOLT Id      a,[PUNTAT.w]
```

En efecto, la variable **PUNTAT** está encerrada entre corchetes [], indicando un direccionamiento **indirecto**. “.w “(word, 2 bytes) identifica el modo **largo**.

Con este direccionamiento se toman en consideración el byte contenido en la dirección de la variable **PUNTAT** y el **byte siguiente**. Dado que cargamos en estas direcciones los valores **FCh** y **07h**, en el registro **A** se almacena el valor presente en la dirección de memoria **FC07h**, es decir **02h**.

Para que quede claro lo anteriormente expuesto volvemos a recordar la declaración de la tabla en la dirección **FC07h**, así se puede constatar fácilmente que en **A** se carga el primer valor asociado a la dirección, es decir **02h**:

(FC07h) TB_02 DC.B 02h,04h,06h,08h

El resto de instrucciones de la subrutina procesarán este dato, llegando a la última instrucción:

(FBD4h) ret

que devuelve el control al programa principal, ejecutándose las instrucciones siguientes que cargan en los dos bytes reservados a la variable **long PUNTAT** el valor **FC03h**:

```
(FA23h)    ld    a,#TB_01.h
(FA25h)    ld    PUNTAT,a
(FA28h)    ld    a,#TB_01.l
(FA2Bh)    ld    {PUNTAT+1},a
(FA2Eh)    call  CALCOLT
```

Mediante la ejecución de la subrutina **CALCOLT** el valor almacenado en el acumulador **A** es **01h**:

(FC03h) TB_01 DC.B 01h,03h,05h,07h

Las instrucciones siguientes:

```
(FA30h)    ld    a,#TB_03.h
(FA32h)    ld    PUNTAT,a
(FA35h)    ld    a,#TB_03.l
(FA38h)    ld    {PUNTAT+1},a
(FA3Bh)    call  CALCOLT
```

cargan en los dos bytes reservados a la variable **long PUNTAT** el valor **FC0Ch**. Ahora, mediante la ejecución de la subrutina **CALCOLT**, el valor almacenado en el acumulador **A** es **10h**:

(FC0Ch) TB_03 DC.B 10h,20h,30h,40h

Resumiendo, se utiliza un direccionamiento **indirecto largo (long)** cuando se reserva al operando un área de memoria de **2 bytes** de longitud.

INDEXADO INDIRECTO LARGO (LONG)

En el ejemplo anterior hemos “apuntado” a tablas diferentes y con la misma subrutina hemos obtenido el **primer valor** de cada tabla. Si hubiéramos querido utilizar el **segundo**, el **tercero** o el **cuarto** valor ... ¿Qué tendríamos que hacer?

Muy sencillo: utilizar el modo **Indexado Indirecto Largo**. Se trata de un modo de direccionamiento análogo al anterior, pero con la inclusión de un **registro índice**, de ahí el

término **indexado**. Introduciendo un valor en este registro podemos responder a la pregunta planteada anteriormente.

Por sencillez vamos a reutilizar el ejemplo anterior, que reproducimos a continuación modificando únicamente la subrutina **CALCOLT**.

En primer lugar definimos una **variable**:

(0094h) PUNTAT DS.B 2

A continuación definimos las **tablas** que contienen los valores: **3 tablas**, cada una con **4 elementos** y con valores predefinidos diferentes.

```
(FC03h) TB_01 DC.B 01h,03h,05h,07h
(FC07h) TB_02 DC.B 02h,04h,06h,08h
(FC0Ch) TB_03 DC.B 10h,20h,30h,40h
```

Ahora escribimos las **instrucciones** del programa de ejemplo:

```
(FA13h)    ld    a,#TB_02.h
(FA15h)    ld    PUNTAT,a
(FA18h)    ld    a,#TB_02.l
(FA1Bh)    ld    {PUNTAT+1},a
(FA1Eh)    call  CALCOLT
.....
.....
(FA23h)    ld    a,#TB_01.h
(FA25h)    ld    PUNTAT,a
(FA28h)    ld    a,#TB_01.l
(FA2Bh)    ld    {PUNTAT+1},a
(FA2Eh)    call  CALCOLT
.....
.....
(FA30h)    ld    a,#TB_03.h
(FA32h)    ld    PUNTAT,a
(FA35h)    ld    a,#TB_03.l
(FA38h)    ld    PUNTAT+1},a
(FA3Bh)    call  CALCOLT
.....
.....
(FB51h)    CALCOLT ld    x,#0FFh
(FB53h)    LOOPX  inc    x
(FB54h)    ld    a,([PUNTAT.w],x)
.....
.....
(FBD0h)    cp    x,#3
(FBD2h)    jrne  LOOPX
(FBD4h)    ret
```


Como ya hemos explicado, después de ejecutar las instrucciones:

```
(FA13h)      ld      a,#TB_02.h
(FA15h)      ld      PUNTAT,a
(FA18h)      ld      a,#TB_02.l
(FA1Bh)      ld      {PUNTAT+1},a
(FA1Eh)      call   CALCOLT
```

La variable **PUNTAT** “contiene” la dirección de **TB_02**, es decir **FC07h**.

Cuando se lanza la subrutina:

```
(FB51h)  CALCOLT ld      x,#0FFh
```

en el registro índice **X** se carga el valor **255 (FFh)** que, como ya sabéis, es el valor máximo que puede contener un registro de **8 bits**.

Con la instrucción:

```
(FB53h)  LOOPX  inc      x
```

se incrementa en **1** el valor contenido en el registro **X**. Además esta instrucción está identificada con la etiqueta **LOOPX**. Puesto que el registro **X** contiene el máximo valor permitido, es decir **255**, este incremento hace que el valor del registro **X** pase a **0**, además de ponerse a **1** el **flag de Acarreo**. La instrucción siguiente:

```
(FB54h)      ld      a,([PUNTAT.w],x)
```

es una instrucción que utiliza el **modo indirecto** (caracterizado por el uso de **corchetes**), **indexada** (utiliza el registro índice **X** encerrado entre **paréntesis**) y de tipo **largo** (. **w**). Con este modo de direccionamiento la dirección de la variable constituye el **desplazamiento (offset)** al que se le añade el valor contenido en el registro **X**.

La **primera vez** que se ejecuta la **subrutina** en el acumulador **A** se carga el valor contenido a la dirección de **[PUNTAT.w]**, es decir **FC07h**, que, como hemos explicado, es **02h**, más el valor del registro **X**, que es **0**. En resumen, la primera vez se almacena en **A** el valor **02h**.

Hemos indicado “primera vez” ya que las instrucciones siguientes:

```
(FBD0h)      cp      x,#3
(FBD2h)      jrne   LOOPX
```

ejecutan la subrutina hasta que el valor contenido en el registro **X** es diferente de **3**. En efecto primero se **compara (cp)** con **3** y si es diferente

el programa **salta (jrne)** a la etiqueta **LOOPX**, donde se **incrementa (inc)** en **1** el registro **X**.

NOTA: La instrucción **jrne** es el acrónimo de **Jump Relative if Not Equal**, es decir salta si no es igual.

Cada vez que se produce un salto a la etiqueta **LOOPX** se añade a la dirección del desplazamiento, que es siempre **FC07h**, un valor de **X incrementado**. Por tanto, en el acumulador **A** se van cargando **todos los valores de la tabla**, como detallamos a continuación:

Recordemos que la definición de la tabla **TB_02** fue:

```
(FC07h)  TB_02  DC.B  02h,04h,06h,08h
```

Por tanto, como ya hemos expuesto, la primera vez que se ejecuta la instrucción:

```
(FB53h)  LOOPX  inc      x
```

X = 0

FC07h + 0

se carga en **A** el valor **02h**.

La segunda vez:

```
(FB53h)  LOOPX  inc      x
```

X = 1

FC07h + 1

se carga en **A** el valor **04h**.

La tercera vez:

```
(FB53h)  LOOPX  inc      x
```

X = 2

FC07h + 2

se carga en **A** el valor **06h**.

La cuarta y última vez:

```
(FB53h)  LOOPX  inc      x
```

X = 3

FC07h + 3

se carga en **A** el valor **08h**.

Cuando el registro **X** contiene el valor **3** la condición de salto **no** se cumple, por lo tanto el programa continúa con la instrucción siguiente:

```
(FBD4h)      ret
```

lo que provoca la salida de la subrutina y la vuelta al programa principal, donde se ejecutan instrucciones similares para las tablas **TB_01** y **TB_03**.

RESUMEN

Direccionamiento Indirecto

En las instrucciones que utilizan direccionamiento **indirecto** la variable de acceso a memoria **no** contiene el **operando** sino la **dirección** que contiene el **operando**, escribiéndose siempre entre **corchetes []**.

En estas instrucciones el operando puede ser tanto la **fuentes** como el **destino** del resultado de la instrucción.

En modo **corto (short)** se puede acceder a una dirección de memoria incluida entre **00h** y **FFh**. En modo **largo (long)**, caracterizado por la presencia de “.w”, se puede acceder a una dirección de memoria incluida entre **0000h** y **FFFFh** (ver Tabla N.1).

Direccionamiento Indexado Indirecto

Este modo de direccionamiento es similar al **indirecto** con la diferencia de que además utiliza los **registros índice X** e **Y** para acceder a una dirección de memoria.

En las instrucciones con este modo de direccionamiento, los **paréntesis ()** situados en los extremos del operando, que puede ser tanto fuente como destino, indican la utilización del modo **indexado**, mientras que los **corchetes []** situados en los extremos de la variable indican la utilización del modo **indirecto**.

Para direccionar en este modo se utiliza una **variable (desplazamiento)** y un **registro índice (X o Y)**, separados por una **coma**.

Se trabaja en modo **corto (short)** cuando la dirección del **desplazamiento** es de **1 byte**. Al sumar el valor contenido en el registro **X** o **Y** la

instrucción puede direccionar un área de memoria incluida entre **00h** y **1FEh**.

Se trabaja en modo **largo (long)** cuando la dirección del **desplazamiento** es de **2 bytes (1 word)**, estando presente la indicación “.w”. Al sumar el valor contenido en el registro **X** o **Y** la instrucción puede direccionar un área de memoria incluida entre **0000h** y **FFFFh**.

Ejemplos de Códigos de Operación

Como colofón a este artículo presentamos algunos ejemplos de instrucciones en **formato Assembler** y en **formato ejecutable**, es decir el **código de operación (op-code)**. Las abreviaturas que hemos utilizado son las mismas que las presentes en los manuales de los fabricantes de micros ST7. Su significado se presenta a continuación.

[short] indirecto corto
[long] indirecto largo
([short],x) . . indexado indirecto corto
([long],x) . . indexado indirecto largo

MODO	INSTRUCCIÓN	OP-CODE
[short]	ld a,[punt]	92 B6 80
[long]	ld a,[punt.w]	92 C6 80
([short],x)	ld a,([punt],x)	92 E6 80
([long],x)	ld a,([punt.w],x)	92 D6 80
([short],y)	ld a,([punt],y)	91 E6 80
([long],y)	ld a,([punt.w],y)	91 D6 80

92 indica **indirecto / indexado indirecto** con **X**

B6 indica modo **corto (short)**

C6 indica modo **largo (long)**

E6 indica modo **indexado corto (short)**

D6 indica modo **indexado largo (long)**

91 indica **indexado indirecto** con **Y**

80 Dirección de ejemplo de la variable **punt**

TABLA N.1

Modo	Ejemplo de formato		Memoria direccionada	Desplazamiento
Indirect Short	ld a,[0E4h]	ld a,[pippo]	00h-FFh	1 Byte
Indirect Long	ld [3Ch.w],a	ld [pippo.w],a	0000h-FFFFh	1 Word
Indirect Indexed Short	ld ([96h],x),a	ld ([pippo],x),a	00h-1FEh	1 Byte
Indirect Indexed Long	ld ([3Ch.w],x),a	ld ([pippo.w],x),a	0000h-FFFFh	1 Word

En esta tabla hemos incluido dos ejemplos de cada uno de los modos de direccionamiento tratados en este artículo, incluyendo el área de memoria que se puede direccionar con cada modo.

TABLA N.2 INSTRUCCIONES Y DIRECCIONAMIENTO (TERCERA PARTE)

Mnemo Instrucción	Descripción Instrucción	Direccionamiento			
		[short]	[long]	[(short),X]	[(long),X]
ADC	Addition with Carry	•	•	•	•
ADD	Addition	•	•	•	•
AND	Logical And	•	•	•	•
BCP	Logical Bit compare	•	•	•	•
BRES	Bit reset	•			
BSET	Bit set	•			
BTJF	Bit test and Jump if false	•			
BTJT	Bit test and Jump if true	•			
CALL	Call subroutine	•	•	•	•
CALLR	Call subroutine relative				
CLR	Clear	•		•	
CP	Compare	•	•	•	•
CPL	One Complement	•		•	
DEC	Decrement	•		•	
HALT	Halt				
INC	Increment	•		•	
IRET	Interrupt routine return				
JP	Absolute Jump	•	•	•	•
JRA	Jump relative always				
JRT	Jump relative				
JRF	Never Jump				
JRIH	Jump if Port INT pin = 1				
JRIL	Jump if Port INT pin = 0				
JRH	Jump if H = 1				
JRNH	Jump if H = 0				
JRM	Jump if I = 1				
JRNM	Jump if I = 0				
JRMI	Jump if N = 1 (minus)				
JRPL	Jump if N = 0 (plus)				
JREQ	Jump if Z = 1 (equal)				
JRNE	Jump if Z = 0 (not equal)				
JRC	Jump if C = 1				
JRNC	Jump if C = 0				
JRULT	Jump if C = 1				
JRUGE	Jump if C = 0				
JRUGT	Jump if (C + Z = 0)				
JRULE	Jump if (C + Z = 1)				
LD	Load	•	•	•	•
MUL	Multiply				
NEG	Negate (2's complement)	•		•	
NOP	No operation				
OR	Or operation	•	•	•	•
POP	Pop from the Stack				
POP	Pop CC				
PUSH	Push onto the Stack				
RCF	Reset carry flag				
RET	Subroutine return				
RIM	Enable Interrupts				
RLC	Rotate left true C	•		•	
RRC	Rotate right true C	•		•	
RSP	Reset stack pointer				
SBC	Subtract with Carry	•	•	•	•
SCF	Set carry flag				
SIM	Disable interrupts				
SLA	Shift left Arithmetic	•		•	
SLL	Shift left Logic	•		•	
SRA	Shift right Arithmetic	•		•	
SRL	Shift right Logic	•		•	
SUB	Subtraction	•	•	•	•
SWAP	Swap nibbles	•		•	
TNZ	Test for Neg & Zero	•		•	
TRAP	S/W trap				
WFI	Wait for interrupt				
XOR	Exclusive OR	•	•	•	•

En la Tabla N.2 se muestran las instrucciones que pueden direccionarse con los modos de direccionamiento indirecto e indexado indirecto, tanto en modo corto (short) como en modo largo (long).



Como en algunas ocasiones este proyecto ha sido desarrollado como consecuencia de vuestras peticiones y sugerencias. En concreto, algunos lectores nos han preguntado sobre dispositivos para ahorrar en la factura mensual de la compañía eléctrica y por sistemas de protección eléctrica para los dispositivos que disponen de la función “stand by”.

AUTO SWITCH para

En ocasiones las ideas para **nuevos proyectos** surgen leyendo las cartas y los correos electrónicos de los lectores que, en ocasiones, son el germen de un proyecto lo suficientemente interesante para desarrollar un kit y publicarlo en la revista.

En una de las cartas que ha llegado a nuestra redacción un lector nos ha contado que, por culpa de la play-station de su hijo, se ha podido **incendiar** su **piso**.

Un tarde, mientras su hijo estaba estudiando, vio salir **humo** del **pequeño alimentador** conectado a la toma red, llegando a **incendiarse** y a propagar el **fuego** al escritorio de madera situado a su lado. Una vez pasado

el peligro se dio cuenta que el alimentador, alojado en un pequeño contenedor de plástico, estaba conectado **día y noche** a la tensión de **red** de **230 V**.

Esta situación podría haberse producido cuando no había **nadie en casa** o, peor aún, durante la **noche** ya que el **daño** podría haber sido **irreparable**.

Nuestro lector, que nos ha pedido no revelar su nombre, nos ha precisado que en su piso también tiene instalado el **Detector** para **fugas de Gas LX.1216**, publicado en la revista **N.137**, para no correr el riesgo de accidentes causados por pérdidas de gas. También nos ha preguntado si disponemos de un kit para

desconectar de la toma red, especialmente por la noche, todos los dispositivos provistos de la función **stand by** y para evitar **cortocircuitos**.

Nuestra respuesta fue que nosotros no hemos publicado ningún circuito de estas características. Al tratarse de un tema muy interesante hemos creído que podíamos desarrollar un kit y hacer partícipes a nuestros lectores. Así nació el kit **LX.1589**, al que hemos denominado **Auto Switch**.

Los dispositivos que disponen de la función de **stand by**, cuando esta función está activa no están totalmente desconectados de la red, ya que los podemos **encender** y **apagar** con un **mando de distancia** y, por tanto, hay que alimentar permanentemente al receptor del mando. Son varios los **dispositivos** que tienen esta característica:

- **Televisores.**
- **Decodificadores TV digital.**
- **Videograbadores.**
- **Videoconsolas.**
- **Lectores DVD.**
- **Cadenas musicales.**

Cuando los dispositivos están en **stand by** permanecen conectados día y noche a la **red eléctrica**. Aunque su absorción de corriente se reduce a un **2%-10%** de la corriente absorbida en funcionamiento, es importante tener presente que su **alimentador** puede llegar a **sobrecalentarse**.

Cuando nuestro **Auto Switch** determina que hemos apagado el aparato con el **mando a distancia** lo **desconecta totalmente** de la toma **red** de **230 voltios**. En estas condiciones podemos dormir tranquilamente por la noche y salir de casa sin preocuparnos.

Hay que tener presente que en **stand by** un **televisor** suele consumir unos 4-5 vatios por hora, si también permanece en **stand by** el **decodificador TV**, el **videograbador**, etc. el **derroche** de corriente eléctrica puede llegar a ser muy considerable.

El **Auto Switch** que aquí presentamos **reduce** la **probabilidad de incendio** por cortocircuito y además **reduce** la **cuenta mensual** que pagamos a la compañía de suministro eléctrico, llegando, en un momento dado, a **amortizarse** el coste del propio dispositivo.

cortocircuitos en la red



FUNCIONAMIENTO del AUTO SWITCH

Nuestro dispositivo permite desconectar **automáticamente** de la red un aparato con **stand by**, como un televisor.

El circuito funciona controlando la **corriente** absorbida por el aparato, que se **reduce** notablemente cuando pasamos de funcionamiento **normal** a **stand by**.

Midiendo la corriente absorbida por un pequeño **núcleo de ferrita** situado junto a un **sensor Hall**, el circuito deja de excitar un **relé**, **separando completamente** el aparato de la **red** en cuanto se baja por debajo del umbral preestablecido.

La conexión del televisor se restablece posteriormente **encendiendo** a través del mando de distancia.

En los ejemplos nos solemos referir a un televisor ya que es el aparato que más comúnmente dispone de la función **stand by**.

ESQUEMA ELÉCTRICO

Comenzamos la descripción del esquema eléctrico por el pequeño **núcleo** de **ferrita** situado junto a un **sensor Hall lineal**, cuya función es medir la **corriente** absorbida por la carga.

Como se puede apreciar en el esquema de la Fig.1, en la sección central del núcleo hay envueltas unas **espiras** de cable que quedan conectadas en serie con la **carga** (ver **L1**).

Cuando la **corriente alterna** absorbida por la **carga** atraviesa las espiras se produce un campo magnético, que es amplificado por el **núcleo** de ferrita. Como consecuencia la superficie magneto-sensible del **sensor Hall (IC1)** queda **excitada**, generando una señal eléctrica con un valor de **tensión** proporcional al valor de la **corriente** que atraviesa las espiras.

Esta **tensión** se aplica a la entrada **inversora** del amplificador operacional **IC2/A**. Una vez amplificada la señal se manda a la entrada **inversora** de **IC2/B**, que junto a los diodos **DS1-DS2**, al condensador **C5** y al integrado **IC2/C**, constituyen un circuito **rectificador de doble semionda**.

La función de este circuito es proporcionar en el terminal de salida (1) del integrado **IC2/C** una tensión **continua** con un valor proporcional a la corriente **alterna** absorbida por la carga.

Esta tensión continua se aplica a la entrada **no inversora** del integrado **IC2/D** (terminal 5), que es utilizado como **comparador de tensión** cuyo umbral de tensión se fija con el trimmer **R12**.

Cuando la tensión presente en la entrada del comparador es **inferior** a la tensión de **umbral** hay una tensión de **0 voltios** en la **salida** al

comparador. En cambio, si la tensión de entrada al comparador es **superior** a la tensión de **umbral** hay una tensión positiva de **12 voltios** en la **salida** del comparador, tensión que provoca el encendido del diodo LED **DL1**.

La señal presente en la salida del comparador se envía a los terminales **5** y **6** de la puerta **NAND IC3/A**, utilizada como **inversor**. Su salida se conecta al terminal **2** de la puerta **NAND IC3/D** y al terminal **9** de la **NAND IC3/C**. Como se puede observar en la Fig.1, el terminal **10** de la puerta **IC3/C** se conecta al terminal **12** de la puerta **IC3/B**, formando entre ambas un **flip-flop**.

La salida del flip-flop (terminal **10** de **IC3/C**) se aplica al terminal **1** de la puerta **NAND IC3/D**. Los niveles lógicos presentes en sus dos entradas controlan, a través del transistor **NPN BC.547 (TR1)**, el **RELE'1** de **12 voltios**, que procede a conectar o a desconectar la tensión a la **carga**.

El diodo LED **DL2**, conectado a la bobina del relé, permite visualizar su estado.

La descripción del circuito eléctrico se completa con el alimentador de **12 voltios**, cuyo primer componente, el transformador **T1**, reduce la tensión de red a **17 voltios AC**. La señal presente en su secundario se rectifica a través del puente **RS1** y se nivela con el condensador **C13** de **1.000 microfaradios**.

La tensión continua de unos **22 voltios** presente en bornes del condensador se aplica al regulador de tensión **L.7812 (IC4)**, que proporciona en su salida una tensión estabilizada de **12 voltios positivos**, utilizada para alimentar los **operacionales**, los integrados **MOS** y el **relé**.

FUNCIONAMIENTO del CIRCUITO

Una vez analizado el esquema eléctrico del circuito vamos a exponer una breve descripción del funcionamiento del circuito.

Partimos de la situación inicial en la que tanto la **carga** como el **Auto Switch** están desenchufados de la red.

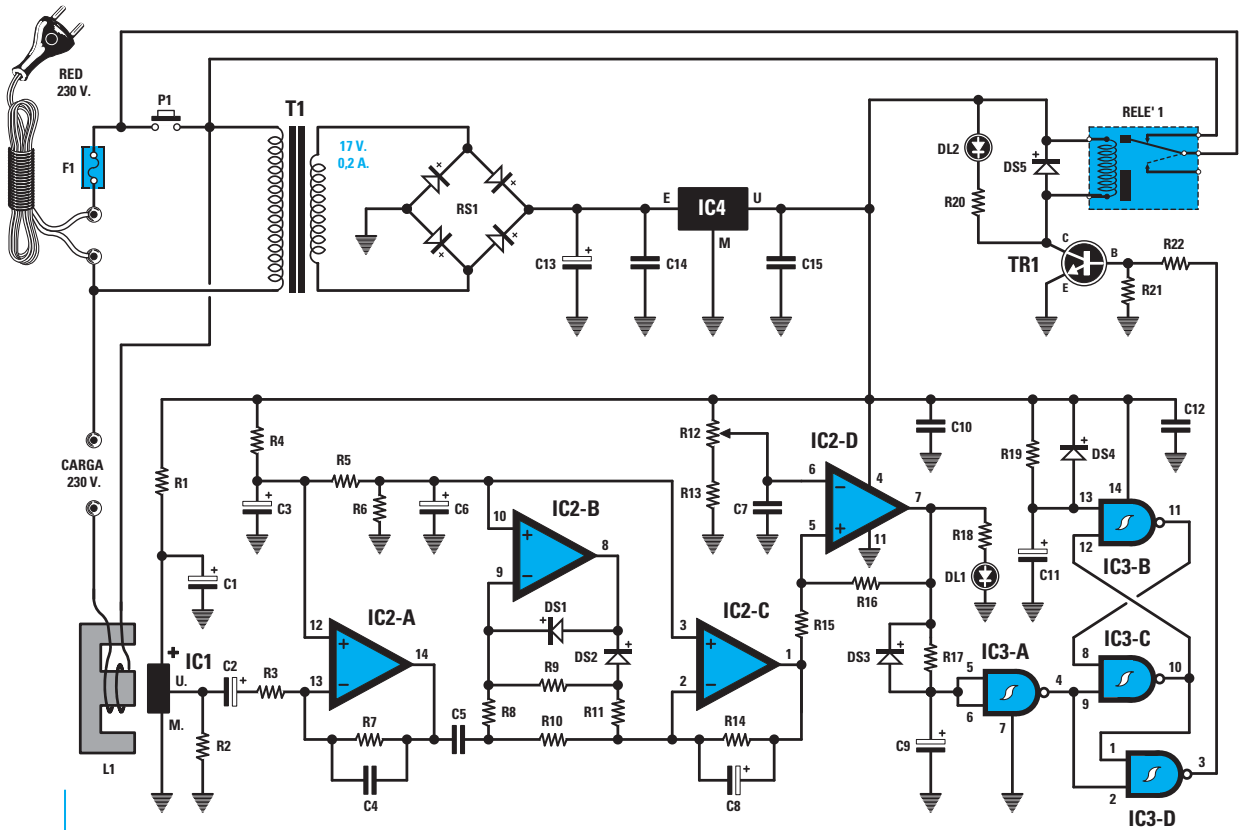


Fig.1 Esquema eléctrico del Auto Switch. La corriente absorbida por la carga atraviesa las dos espiras envueltas sobre un pequeño núcleo de ferrita que está en contacto con un sensor Hall. La superficie del sensor, excitada por el campo magnético generado por el núcleo, produce una tensión proporcional a la corriente que atraviesa las espiras.

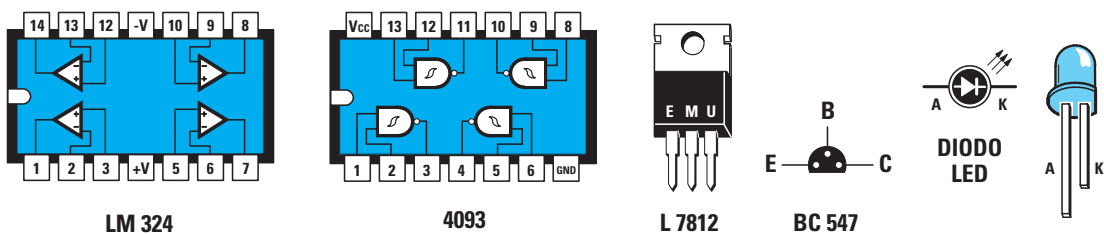


Fig.2 Conexiones de los componentes necesarios para realizar el Auto Switch. Las conexiones de los integrados LM.324 y 4093 se muestran vistas desde arriba y con la muesca de referencia en forma de U orientada hacia la izquierda. El integrado estabilizador L.7812 se muestra frontalmente, mientras que las conexiones del transistor BC.547 se muestran vistas desde abajo.

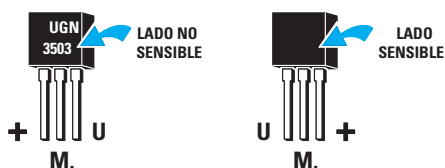


Fig.3 Conexiones del integrado UGN.3503. El lado sensible, que debe ponerse en contacto con el núcleo de ferrita, es reconocible por la ausencia de indicaciones serigráficas.

Al conectar la clavija de red, la tensión sobre la carga no cambia, ya que al no accionar el pulsador **P1** no llega tensión al transformador **T1** ni, por tanto, al **RELE'1**, que, al no quedar excitado, no proporciona tensión a la carga.

Si accionamos el pulsador **P1** el transformador **T1** queda alimentado, proporcionando a todo el circuito la tensión de **12 voltios**.

Observando el circuito formado por el **flip-flop IC3/B-IC3/C** y por la puerta **NAND IC3/D**, conectada al transistor **TR1**, se puede ver como el condensador **C11**, descargado inicialmente, fuerza a **nivel lógico 0** el terminal **13** de **IC3/B** al que está conectado.

Por otra parte, el condensador **C9**, conectado a los terminales **5** y **6** de la **NAND IC3/A**, se descarga, generando en el terminal **4** de **IC3/A** un **nivel lógico 1**.

En estas condiciones el **flip-flop** genera un nivel lógico **0** en su terminal de salida (**10**), conectado al terminal **1** de la **NAND IC3/D**. Esta a su vez genera un nivel lógico **1** en su terminal de salida (**3**) conectado a la base del transistor **TR1**, que, puesto que entra en conducción, **excita al RELE'1**.

De esta forma se proporciona tensión a la carga y al mismo tiempo al transformador **T1**, así que

aunque dejemos de presionar el **pulsador P1** se mantiene la alimentación.

La corriente absorbida por un televisor cuando está funcionando normalmente provoca una tensión **positiva** en el terminal **1** de **IC2/C** que, siendo superior al valor de umbral prefijado por **R12**, genera en el terminal de salida del **comparador (7)** una tensión **positiva** que enciende el diodo LED **DL1** y produce un nivel lógico **0** en el terminal **4** de **IC3/A**, conectado a la entrada del **flip-flop**.

Esta situación provoca la **conmutación** del **flip-flop**, llevando a nivel lógico **1** a su salida (terminal **10**).

Ya que en el terminal **4** de la puerta **NAND IC3/A** hay un nivel lógico **0**, la salida de la puerta **NAND IC3/D** (terminal **3**) pasa a nivel lógico **1**, por lo que se mantiene **excitado** el **RELE'1** alimentando así la carga.

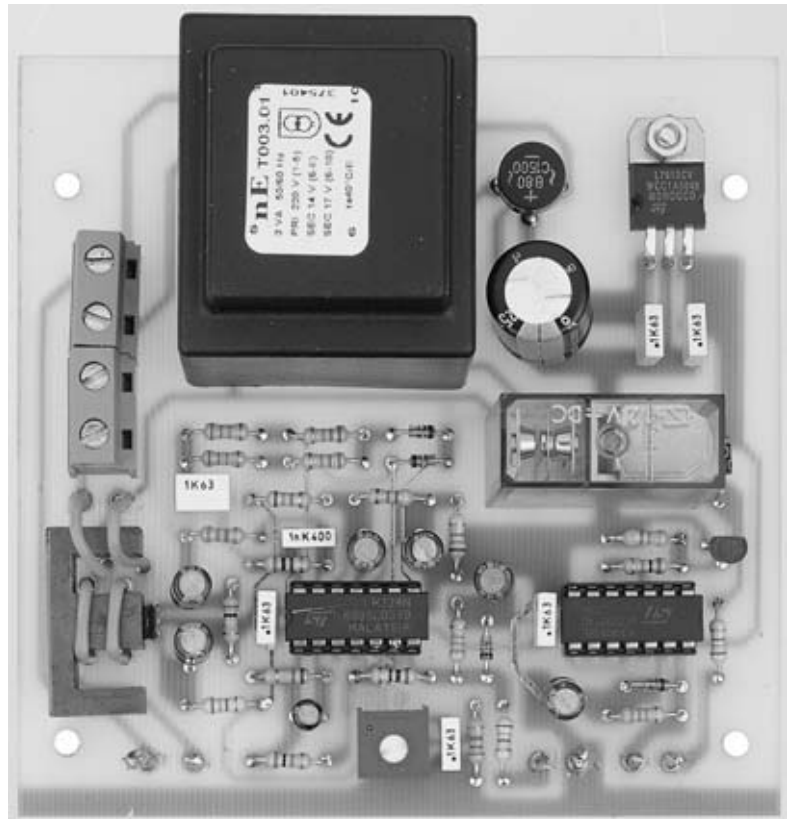
Supongamos que ahora se apaga el televisor con el **mando a distancia** llevándolo así a **stand by**.

La corriente bajará bruscamente por debajo del umbral fijado, de este modo la tensión en el terminal **7** del comparador **IC2/D** pasa a ser **0 voltios**.



Fig.4 En el panel posterior, que se proporciona perforado, hay que instalar la toma de salida para la carga, el portafusibles y el cable de red con su correspondiente goma pasacables.

Fig.5 Fotografía del circuito impreso del Auto Switch con todos sus componentes montados. Sobre el núcleo de ferrita hay que envolver dos espiras haciendo pasar el cable por los agujeros practicados en el circuito impreso con este propósito.



Como consecuencia el LED **DL1** se apaga, señalando así la desconexión de la carga de la red. En el terminal **4** de la **NAND IC3/A** hay un nivel lógico **1** que, junto al nivel lógico **1** presente en el terminal de salida del **flip-flop (10)**, provocan un nivel lógico **0** en el terminal de salida de la **NAND IC3/D**, poniendo en corte al transistor **TR1**. En estas circunstancias el **RELE'1** no queda **excitado**, lo que provoca la **desconexión** del transformador **T1** de la red, volviendo así a la situación de la que partíamos inicialmente.

REALIZACIÓN PRÁCTICA

Siguiendo las indicaciones que exponemos a continuación el montaje de este circuito no presenta ninguna dificultad.

Para realizar el **Auto Switch** se precisa el circuito impreso de doble cara **LX.1589**. En este impreso se han de montar todos los componentes mostrados en la Fig.6.

Aconsejamos instalar en primer lugar los **2 zócalos** para los integrados **IC2** e **IC3**,

respetando la referencia de la serigrafía del circuito impreso y teniendo cuidado, al soldar los terminales, en no provocar cortocircuitos. El zócalo del integrado **IC2** se monta con su muesca de referencia orientada hacia la **izquierda**, mientras el zócalo del integrado **IC3** se monta con su muesca de referencia orientada hacia la **derecha** (ver Fig.6).

El montaje puede continuar con las **resistencias**, todas de **1/4 vatio**, identificando su valor a través de las **franjas de color** presentes en su cuerpo.

Después de montar las resistencias podemos continuar con los **4 diodos** de silicio **1N.4148**, instalándolos en los agujeros correspondientes a las referencias **DS1-DS2-DS3-DS4** y orientando sus franjas de color **negro** como se muestra en la Fig.6.

A continuación hay que instalar el **diodo** de silicio **1N.4007** en los agujeros correspondientes a la referencia **DS5**, orientando hacia **arriba** su franja de color **blanco** (ver Fig.6).

Ahora se puede proceder a la instalación de los **condensadores de poliéster**, controlando el valor serigrafiado, y de los **condensadores electrolíticos**, respetando la polaridad de sus terminales y teniendo presente que su terminal **más largo** es el **positivo (+)**.

Completada esta operación hay que montar el trimmer **R12 (10.000 ohmios)**, el **RELE'1** de **12 voltios** y el integrado **IC4**, doblando en este caso sus terminales en forma de **L** y fijando su cuerpo al circuito impreso con un tornillo y su correspondiente tuerca.

Es el momento de montar el puente rectificador **RS1**, respetando la polaridad de sus terminales **+/-** y separando su cuerpo del circuito impreso unos **5-6 milímetros**.

Seguidamente hay que proceder a la instalación del transistor **TR1**, orientando la parte **plana** de su cuerpo hacia el **RELE'1**.

Llegado este punto se puede proceder con el montaje del **núcleo de ferrita L1** y del **sensor Hall IC1**, teniendo en cuenta las

consideraciones que se exponen a continuación.

Al introducir el integrado **IC1** en los agujeros correspondientes del circuito impreso hay que orientar su **lado sensible**, es decir el lado que **no tiene nada serigrafiado**, hacia la **izquierda** (ver Figs.3 y 6).

Después de soldar los terminales de **IC1** hay que apoyar sobre el integrado el segmento central del núcleo de ferrita en forma de **E**.

Ahora hay que soldar un extremo del trozo de cable incluido en el kit en un agujero (cualquiera) de los dos situados al lado de la clema. A continuación hay que pasar el cable por los **6 agujeros** presentes en el circuito impreso, envolviendo con cierta fuerza el cable de modo que se obtengan **2 espiras** sobre el segmento central del núcleo. Por último hay que soldar el extremo del cable libre en el agujero situado al lado del punto inicial del cable (ver fotografía de la Fig.5). Una vez realizada esta operación el núcleo de ferrita queda mecánicamente unido al circuito impreso.

LISTA DE COMPONENTES LX.1589

R1 = 680 ohmios
R2 = 100.000 ohmios
R3 = 3.300 ohmios
R4 = 6.800 ohmios
R5 = 4.700 ohmios
R6 = 1.800 ohmios
R7 = 1 Megaohmio
R8 = 22.000 ohmios
R9 = 22.000 ohmios
R10 = 22.000 ohmios
R11 = 12.000 ohmios
R12 = Trimmer 10.000 ohmios
R13 = 1.500 ohmios
R14 = 120.000 ohmios
R15 = 10.000 ohmios
R16 = 1 Megaohmio
R17 = 33.000 ohmios
R18 = 680 ohmios
R19 = 33.000 ohmios
R20 = 680 ohmios
R21 = 47.000 ohmios
R22 = 10.000 ohmios
C1 = 10 microF. electrolítico
C2 = 10 microF. electrolítico
C3 = 10 microF. electrolítico
C4 = 1.000 pF poliéster
C5 = 1 microF. poliéster
C6 = 10 microF. electrolítico
C7 = 100.000 pF poliéster

C8 = 2,2 microF. electrolítico
C9 = 10 microF. electrolítico
C10 = 100.000 pF poliéster
C11 = 10 microF. electrolítico
C12 = 100.000 pF poliéster
C13 = 1.000 microF. electrolítico
C14 = 100.000 pF poliéster
C15 = 100.000 pF poliéster
L1 = Ver texto
RS1 = Puente rectificador 100V 1A
DS1 = Diodo 1N.4148
DS2 = Diodo 1N.4148
DS3 = Diodo 1N.4148
DS4 = Diodo 1N.4148
DS5 = Diodo 1N.4007
DL1 = Diodo LED
DL2 = Diodo LED
TR1 = Transistor NPN BC.547
IC1 = Sensor UGN.3503
IC2 = Integrado LM.324
IC3 = Integrado CMOS 4093
IC4 = Integrado L.7812
F1 = Fusible 2,5 amperios
T1 = Transformador 3 vatios (T003.01) sec.
0-14-17 V 0,2 A
RELE'1 = Relé 12V
P1 = Pulsador 250 voltios
NOTA: Todas las resistencias utilizadas en este circuito son de 1/4 vatio.

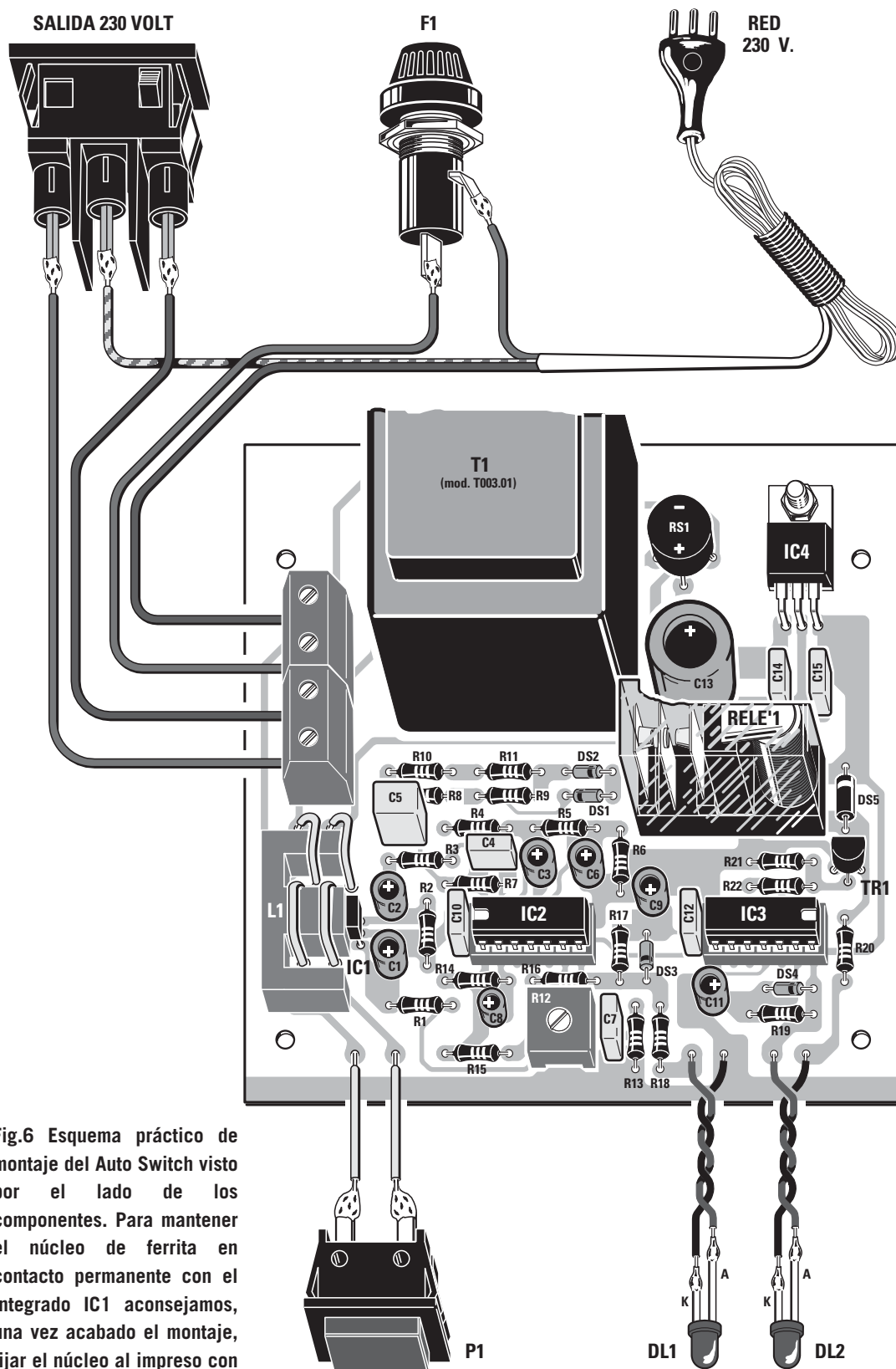


Fig.6 Esquema práctico de montaje del Auto Switch visto por el lado de los componentes. Para mantener el núcleo de ferrita en contacto permanente con el integrado IC1 aconsejamos, una vez acabado el montaje, fijar el núcleo al impreso con un poco de pegamento o de silicona.



Fig.7 una vez instalado el circuito impreso dentro del mueble plástico hay que realizar las conexiones con los componentes exteriores.

Seguidamente hay que situar el núcleo de ferrita de modo que haga un contacto perfecto con el **sensor Hall IC1** (para un mejor contacto se puede aplicar en los lados un poco de pegamento o de silicona una vez situado en posición).

Una vez realizado el montaje de la ferrita y del sensor Hall se puede proceder a instalar **IC2** e **IC3** en sus correspondientes zócalos, respetando la muesca de referencia en forma de **U** y teniendo mucho cuidado en no torcer ningún terminal.

A continuación se pueden montar las dos **clemas** de 2 polos (una utilizada para la

tensión de entrada de **red** y otra para conectar la **carga**) y el transformador **T1**, soldando sus **5** terminales a las pistas del circuito impreso.

Una vez concluido el montaje de los componentes del circuito impreso hay que fijarlo en la base del mueble de plástico con cuatro tornillos.

En el panel frontal (perforado y a serigrafiado) hay que instalar el pulsador **P1** y los **porta leds** para los diodos **DL1** y **DL2**.

En el panel posterior (perforado) hay que montar la **toma** para los **230 voltios** de **salida**

y el **portafusibles**. Una vez montado hay que instalar en su interior un fusible de **2,5 amperios**. En el agujero situado bajo el portafusibles hay que montar la **goma pasacables** para el cable de red (ver Fig.4).

Ahora se pueden conectar los **cables de red** a la **clema** situada en la parte **superior**, conectando en **serie** a uno de ellos el **portafusibles F1**. A la clema **inferior** hay que conectar la **toma de red de 230 voltios** para la **carga**, como se muestra en la Fig.6. La **masa** del cable de red ha de conectarse al **terminal central** de la toma de salida.

Por último hay que realizar la conexión del **pulsador P1** y de los **diodos LED DL1-DL2**, respetando en estos últimos la polaridad de sus terminales (ver Fig.6).

AJUSTE del CIRCUITO

Antes de cerrar el mueble hay que **ajustar** el umbral de intervención del **Auto Switch**. Esta operación ajusta el circuito con el dispositivo que se conecta como carga (**televisor, videograbador, reproductor DVD**, etc.).

Para realizar este ajuste hay que proceder como se indica a continuación:

- Conectar la carga al **Auto Switch**, por ejemplo el televisor.

- Girar completamente hacia la izquierda el **trimmer R12**.

- Conectar el **Auto Switch** a la red y presionar el pulsador **Power**. Se encenderán los diodos **LED Load** y **Relé On** del panel frontal. El televisor está alimentado.

- Ahora hay que poner el televisor en estado **stand by** (con el mando a distancia) y girar lentamente el **trimmer R12** en sentido de las agujas del reloj hasta que se apague el **LED Load**. Al mismo tiempo se apagará también el **LED Relé On**, quedando sin excitar el relé, apagándose el televisor.

Con esta operación termina el ajuste del **umbral de intervención** del Auto Switch.

UTILIZACIÓN: ENCENDIDO

Después de conectar la carga al **Auto Switch**, por ejemplo el televisor, hay que accionar el pulsador **Power** del **Auto Switch**. El **LED Relé On** se ilumina indicando así que el televisor está conectado a la red.

NOTA: Hay que cerciorarse de que el interruptor de encendido del televisor esté en la posición **ON**.

En estas condiciones se encenderá el **LED stand-by** del televisor. Accionando el pulsador **stand-by** del mando a distancia el televisor se enciende normalmente y el **LED Load** del **Auto Switch** se ilumina como confirmación del funcionamiento del circuito de control.

UTILIZACIÓN: APAGADO

Para apagar el televisor hay que pulsar de nuevo el pulsador **stand-by** del mando a distancia. Los diodos **LED Load** y **Relé On** del panel frontal del **Auto Switch** se apagarán y el televisor se desconectará de la red.

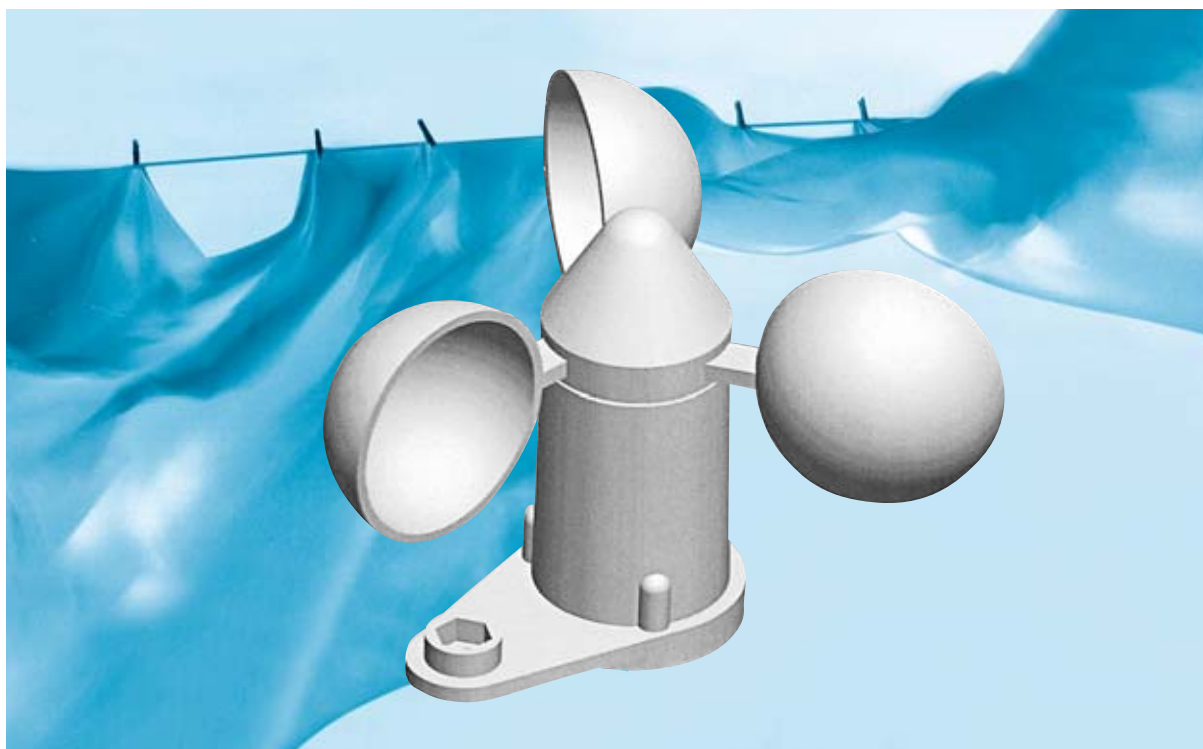
PRECIO de REALIZACIÓN

LX.1589: Precio de todos los componentes necesarios para realizar el circuito **Auto Switch**, incluyendo circuito impreso, integrados, transformador y todos los componentes mostrados en las Figs.5-6, **excluido** únicamente el mueble de plástico **MO.1589****51,85 €**

CC.1589: Circuito impreso**14,80 €**

MO.1589: Precio del mueble plástico **MO.1589** con paneles perforados y panel frontal serigrafiado**15,55 €**

ESTOS PRECIOS NO INCLUYEN I.V.A.



ANEMÓMETRO

Este anemómetro puede ser programado para excitar un relé o para emitir una señal acústica cuando el viento supera una determinada velocidad que podemos ajustar. El relé puede utilizarse para controlar una sirena, un motor o cualquier otro dispositivo eléctrico.

En la revista N.239 hemos presentado una Estación meteorológica que incluye Anemómetro, Anemoscopio, Termómetro y Pluviómetro. Muchos lectores nos han preguntado si disponemos de un **anemómetro independiente**, ya que las otras aplicaciones no les son de utilidad y no están dispuestos a pagar el precio de todos los aparatos cuando solo precisan uno.

Nos han llegado múltiples propuestas de utilización para el anemómetro como justificación para la petición de un **anemómetro independiente**. Ante estas peticiones, de las que seguidamente

relacionamos unas cuantas, hemos desarrollado el kit **LX.1606**.

La primera petición que nos llegó era de grupo de atletas, ya que la utilización del anemómetro es indispensable para establecer la **velocidad del viento** en las **competiciones**.

También nos han llegado peticiones de **socorristas** que nos han señalado que este instrumento es muy útil para detectar la presencia del viento para controlar el uso de las **sombrillas** y para dar información a los usuarios de **tablas de surf** y pequeñas **embarcaciones de vela**.

Una de las aplicaciones más curiosas que nos han propuesto es utilizar un anemómetro para poder **enrollar un toldo de forma automática** cuando la velocidad del viento alcanza cierta intensidad, evitando que se rasgue o que las varillas de sujeción se partan. Evidentemente para realizar esta función es necesario añadir a nuestro circuito un **relé** capaz de controlar un motor eléctrico.

Algunos apasionados a la **vela** han pensado en un anemómetro para controlar la intensidad del viento antes de salir a mar abierto.

Un electricista especializado en ventilación de locales nos ha propuesto utilizar un anemómetro para hacer una **comparativa** entre los **ventiladores** del mercado y determinar cuales más eficaz.

Un suscriptor nos ha comunicado que estaría interesado en instalar un anemómetro equipado con una alarma sonora en su

caravana para evitar que ráfagas de viento de elevada intensidad puedan hacerla volcar en carretera.

Podríamos continuar con la lista de aplicaciones que nos han propuesto los lectores. Ahora bien, llegados a este punto seguramente habréis comprendido que en todos los casos propuestos la solución consiste en un **anemómetro** que incluya una **alarma acústica** y un **relé** que puedan ajustarse para entrar en funcionamiento ante una velocidad del viento que podamos **programar** según nuestras necesidades.

ESQUEMA ELÉCTRICO

Como se puede observar en el esquema eléctrico de la Fig.2, el **anemómetro** dispone de un cordón con **dos cables** que se conectan a los dos terminales de entrada del circuito de control, uno está conectado a la **masa** del circuito y otro al terminal **2** del optoacoplador **OC1**.

PROGRAMABLE



Fig.1 En esta fotografía se muestra el mueble de plástico que incluye en su interior el circuito de control. En su parte exterior se puede observar el display y los pulsadores de programación.

En el interior del **sensor anemométrico SE1** hay un **relé reed** que cortocircuita cíclicamente a **masa** el terminal **2** con el movimiento de los brazos del anemómetro causado por el viento. Cuanto **mayor** sea la **velocidad del viento** más veces por segundo se excitará el relé reed y, por tanto, la señal producida tendrá una **frecuencia mayor**.

Del terminal **5** del optoacoplador **OC1** salen los **impulsos digitales** que se aplican al terminal **6** del microprocesador **IC1**, un **PIC** tipo **16F819** programado para desempeñar todas las funciones necesarias para el **anemómetro**.

La función más importante del microprocesador es mostrar en el doble display la **velocidad del viento** expresada en **Km/hora**.

Para obtener valores muy precisos el microprocesador toma la velocidad en **intervalos regulares** de una decena de

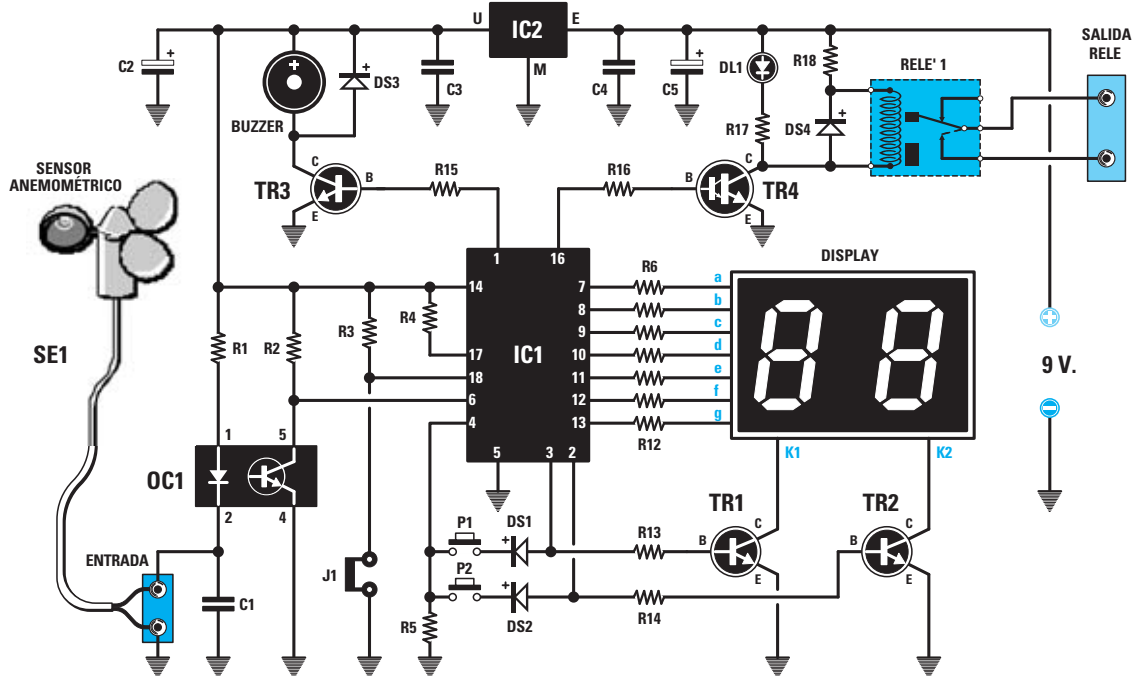


Fig.2 Esquema eléctrico del Anemómetro Programable LX.1606. Para programar la velocidad a la que se desea que se excite el relé y suene el zumbador hay que cortocircuitar los dos terminales del conector J1 con el conector hembra (jumper) mostrado en la Fig.8 y utilizar los pulsadores P1-P2 para fijar el valor.

LISTA DE COMPONENTES LX.1606

R1 = 2.200 ohmios 1/8 vatio	R16 = 10.000 ohmios 1/8 vatio	TR3 = Transistor NPN BC.547
R2 = 10.000 ohmios 1/8 vatio	R17 = 1.000 ohmios 1/8 vatio	TR4 = Darlington NPN BC.517
R3 = 10.000 ohmios 1/8 vatio	R18 = 56 ohmios 1/8 vatio	OC1 = Optoacoplador H11AV/1A
R4 = 10.000 ohmios 1/8 vatio	C1 = 33.000 pF poliéster	IC1 = CPU PIC programada (EP1606)
R5 = 10.000 ohmios 1/8 vatio	C2 = 10 microF. electrolítico	IC2 = Integrado MC.78L05
R6 = 330 ohmios 1/8 vatio	C3 = 100.000 pF poliéster	Display = Display doble LT533
R7 = 330 ohmios 1/8 vatio	C4 = 100.000 pF poliéster	Relé 1 = Relé 6V
R8 = 330 ohmios 1/8 vatio	C5 = 100 microF. electrolítico	Buzzer = Zumbador
R9 = 330 ohmios 1/8 vatio	DS1 = Diodo 1N.4148	J1 = Punte
R10 = 330 ohmios 1/8 vatio	DS2 = Diodo 1N.4148	P1 = Pulsador
R11 = 330 ohmios 1/8 vatio	DS3 = Diodo 1N.4148	P2 = Pulsador
R12 = 330 ohmios 1/8 vatio	DS4 = Diodo 1N.4148	SE1 = Sensor anemométrico SE1.20
R13 = 10.000 ohmios 1/8 vatio	DL1 = Diodo LED	
R14 = 10.000 ohmios 1/8 vatio	TR1 = Transistor NPN BC.547	
R15 = 10.000 ohmios 1/8 vatio	TR2 = Transistor NPN BC.547	

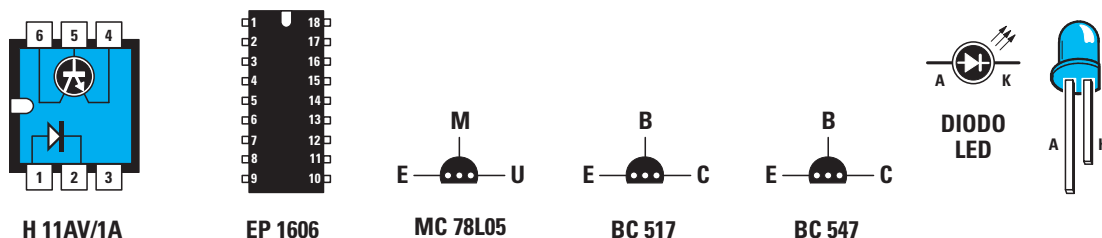


Fig.3 Conexiones, vistas desde arriba, del optoacoplador H.11AV/1A y de la CPU programada EP.1606. Las conexiones de los transistores y del 78L05 se muestran vistas desde abajo.

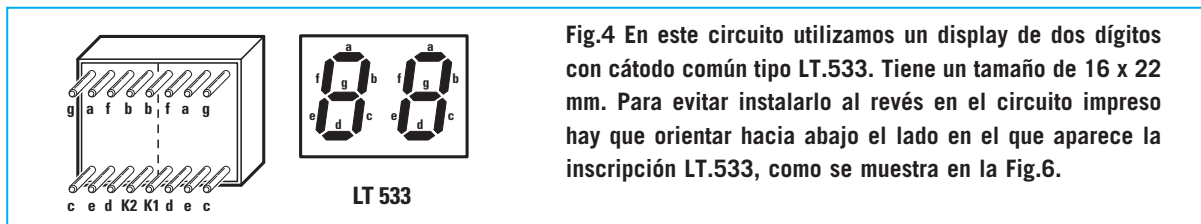


Fig.4 En este circuito utilizamos un display de dos dígitos con cátodo común tipo LT.533. Tiene un tamaño de 16 x 22 mm. Para evitar instalarlo al revés en el circuito impreso hay que orientar hacia abajo el lado en el que aparece la inscripción LT.533, como se muestra en la Fig.6.

segundos, a continuación realiza la **media** y por último lleva el resultado al **display**.

El display está controlado por el microprocesador de forma **multiplexada**, esto es a los segmentos **a-b-c-d-e-f-g** de los dos dígitos llega la tensión **positiva** necesaria para encender los segmentos adecuados, mientras que alternativamente y de forma **muy rápida** los dos transistores **TR1-TR2** cortocircuitan a **masa** el terminal **K1** del primer dígito y luego el terminal **K2** del segundo dígito. Aunque los dos dígitos se enciendan alternativamente, la velocidad de conmutación es tan elevada que nuestros ojos ven **encendidos ambos dígitos**.

Cuando el **anemómetro** mide una **velocidad del viento**, por ejemplo de **15 Km/h**, después de unos pocos segundos el microprocesador hará salir de los terminales **8-9** una **tensión positiva** para alimentar los segmentos **b-c** de ambos dígitos, pero ya que solo se alimenta la **Base** del transistor **TR1** a través del terminal **3**, veremos aparecer el número **1** en el dígito de la **izquierda**.

El microprocesador provoca una tensión positiva de las terminales **7-12-13-9-10** para alimentar los segmentos **a-f-g-c-d**, haciendo llegar el número **5** en ambos dígitos, pero ya que solo se alimenta la **Base** del transistor **TR2**, veremos aparecer el número **5** en el dígito de la **derecha**.

La **velocidad** de conmutación de los dos dígitos es tan **elevada** que nuestros ojos los verán encendidos **simultáneamente**, en este caso con el número **15**.

Los pulsadores **P1-P2**, conectados a las **Bases** de los transistores **TR1-TR2**, sirven para **ajustar** la velocidad del viento a la que deseamos excitar el **relé** (controlado por el transistor **TR4**) y el **zumbador** (controlado por el transistor **TR3**).

El relé puede utilizarse **directamente** para controlar elementos de **pequeña y media potencia** (2 amperios / 30 voltios máximo) y para controlar **relés de potencia** alimentados a **12 voltios**, los que, a su vez, pueden servir para controlar cualquier **dispositivo de potencia** (**motores, bombillas, sirenas, etc.**).

Una vez que se supera la **velocidad máxima** que hemos programado el terminal **16** de **IC1** pasa a **nivel lógico 1**, es decir tiene una tensión positiva que, al controlar la **Base** del transistor **TR4**, lo pone en conducción y, como consecuencia, se **excita** el **relé**.

Automáticamente del terminal **1** sale una señal de **onda cuadrada** con una frecuencia de unos **2 KHz** que, al controlar la **Base** del transistor **TR3**, hace emitir al pequeño **zumbador** conectado a su **Colector** una **señal acústica** para indicar que la **velocidad** del viento ha superado el valor **máximo** que hemos programado.

ALIMENTACIÓN del CIRCUITO

Para alimentar el anemómetro es necesaria una tensión **continua no estabilizada** de unos **9 voltios**, tensión que se puede obtener a través de **pilas** o de un sencillo **transformador** de tensión conectado a la red eléctrica de **230 voltios**.

Aconsejamos utilizar **pilas** únicamente en caso de que necesitéis un anemómetro **portátil** ya que el circuito absorbe, con el relé excitado, una corriente **máxima** de unos **150 miliamperios**. En estas condiciones una pila corriente de **9 voltios** tendría una autonomía en torno a **1 hora**. Para conseguir una autonomía de **8-10 horas** (en consumo máximo) se pueden conectar en serie **dos pilas de petaca** de **4,5 voltios**.

Si utilizáis el anemómetro en un **sitio fijo** la solución más sencilla consiste en utilizar un

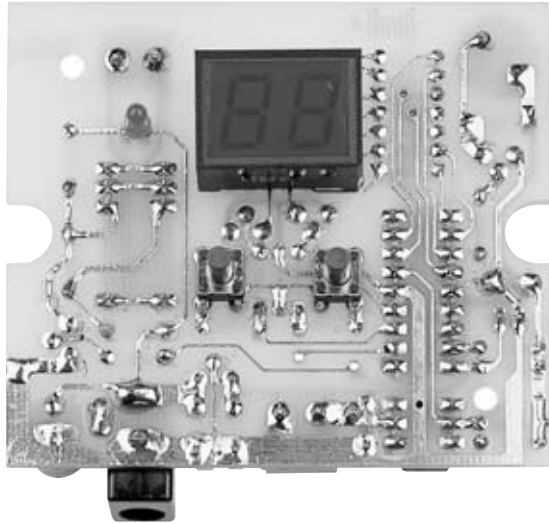
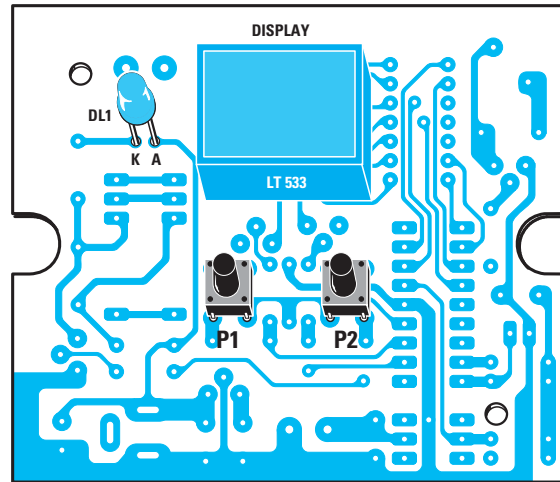


Fig.6 Esquema práctico de montaje del circuito impreso LX.1606 visto por el lado del display y de los pulsadores P1-P2. El diodo LED DL1 hay que montarlo orientando el terminal más largo (Ánodo) hacia el display.

Fig.5 Fotografía del circuito impreso LX.1606 visto por el lado del display y de los pulsadores de programación P1-P2. Para aprender a utilizar estos pulsadores hay que leer el texto del artículo.



pequeño **transformador/reductor** con una potencia de **2-3 vatios**.

REALIZACIÓN PRÁCTICA

Es aconsejable comenzar el montaje instalando, en el lado de la placa mostrado en la Fig.6, el **display LT.533**, el diodo LED **DL1** y los dos pequeños pulsadores **P1-P2**.

Cuando se inserten los terminales del **display** en los agujeros del circuito impreso hay que orientarlo correctamente, teniendo presente que el **lado** del cuerpo que tiene impresa la **referencia LT.533** ha de orientarse hacia los dos pulsadores **P1-P2**. Si no se hace así el display **no** se encenderá.

Si, por error, se instala al revés hay que desoldarlo y girarlo. En cuanto se suelde y se alimente aparecerá el número **0-0**.

Es aconsejable **inicialmente** fijar el diodo LED **DL1 sin soldarlo**. Después de haber colocado el circuito impreso dentro del mueble de plástico, y una vez establecida la **altura** a la que conviene fijarlo para que sobresalga su cabeza por encima del mueble, ya se puede soldar.

El terminal **más largo** del diodo LED (Ánodo) debe orientarse hacia el **display**, como se puede ver en la Fig.6.

Una vez realizada esta operación hay que dar la vuelta al circuito impreso **LX.1606** y, en el lado mostrado en la Fig.8, montar los **zócalos** para el optoacoplador **OC1** y para el microprocesador **IC1** y, a continuación, el **conector de 2 terminales (J1)**, utilizado como **punte** (jumper).

El montaje puede continuar con la instalación de las **resistencias** y de los **diodos de silicio** con cuerpo en vidrio **DS1-DS2-DS3-DS4**, orientando el lado de su cuerpo rodeado por una fina franja **negra** como se muestra en el esquema de montaje práctico de la Fig.8.

Ahora se puede proceder a la instalación de los **condensadores de poliéster (C1-C3-C4)** y de los **condensadores electrolíticos (C2-C5)**, respetando en estos últimos la polaridad de sus terminales y teniendo presente que su terminal **más largo** es el **positivo (+)**.

Es el momento de instalar los transistores **TR1-TR2-TR3-TR4**, orientando el **lado plano** de su

Fig.7 Fotografía del circuito impreso LX.1606 visto por el lado de los componentes. Este circuito impreso debe instalarse dentro del mueble de plástico mediante 2 tornillos (ver Fig.9).

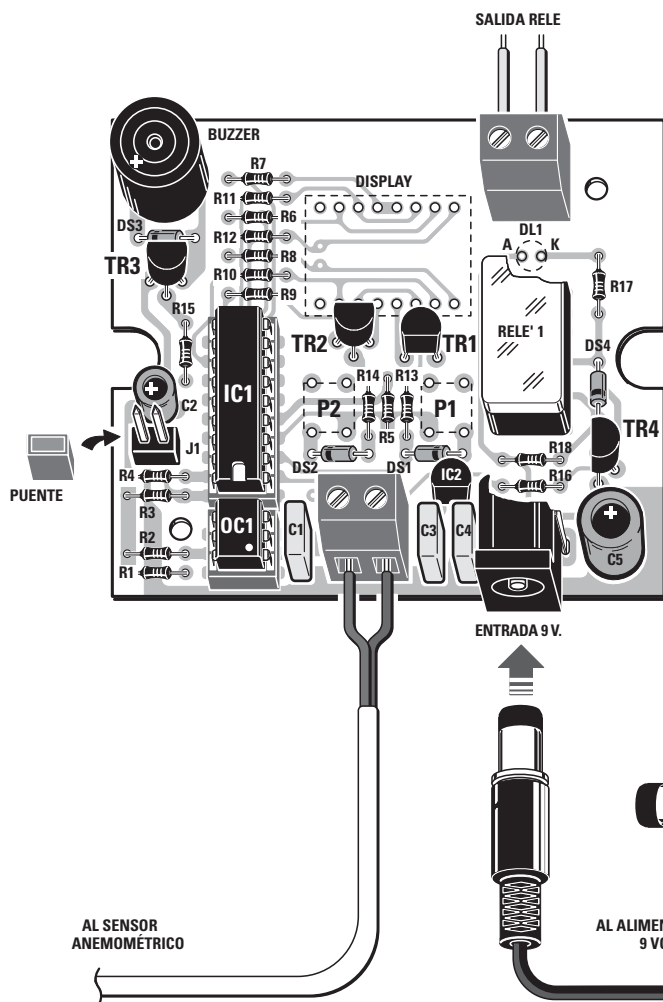
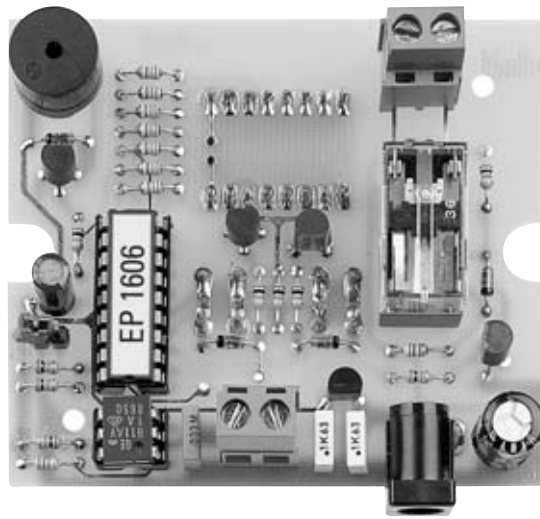
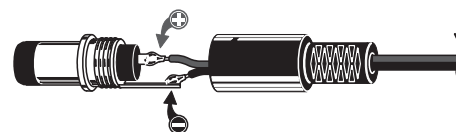


Fig.8 Esquema práctico de montaje del circuito impreso LX.1606 visto por el lado de los componentes. El zumbador (buzzer) debe montarse orientando hacia la parte inferior-izquierda el borne +. En el conector de alimentación de 9 Voltios (ver parte inferior), el cable negativo va conectado al terminal exterior y el cable positivo al terminal central.



cuerpo como se muestra en el esquema de montaje práctico de la Fig.8.

El **cuerpo** de estos transistores no debe tocar el circuito impreso. Al montarlos hay que controlar los terminales de tal forma que haya una distancia de unos **5 milímetros** entre su cuerpo y la superficie del circuito impreso.

A continuación se puede montar el integrado estabilizador **IC2**, componente que tiene la misma forma y tamaño que los transistores. Al montarlo hay que orientar la **parte plana** de su cuerpo hacia los condensadores **C3-C4** (ver Fig.8).

Llegado este punto se puede proceder a montar los componentes de dimensiones mayores, esto es, el **relé**, el **zumbador** (orientando el terminal **+** hacia la parte inferior-izquierda) y las **dos clemas de 2 polos** (una se utiliza para fijar los dos cables que llegan del anemómetro y otra para conectar el dispositivo a controlar con los contactos del **relé**).

Los **contactos del relé** se comportan como un **interruptor**, conectándose en **serie** entre el **dispositivo** a controlar y su tensión de **alimentación**, que puede tener un valor máximo de **30 voltios / 2 amperios**.

Continuando con la instalación hay que montar,

bajo el **relé**, el **conector** para la toma externa de **alimentación de 9 voltios**.

Una vez finalizado el montaje de los componentes se puede proceder a instalar, en sus correspondientes zócalos, el optoacoplador **OC1**, orientando hacia abajo su pequeño **punto** de referencia, y el microprocesador **IC1**, orientando hacia abajo su muesca de referencia en forma de **U** (ver Fig.8).

MONTAJE en el MUEBLE

El circuito impreso debe fijarse dentro de su pequeño mueble de plástico utilizando **2 tornillos**. El mueble incluye dos paneles de plástico **perforados** para acceder al **conector de alimentación** y para la entrada del **cordón del anemómetro**.

EL CORDÓN del ANEMÓMETRO

El anemómetro incluye un cordón con dos cables de **2 metros** de longitud, si se precisa cubrir más distancia hay que alargarlo.

Para alargar el cordón **no** hay que utilizar cable de cobre del mismo diámetro que el incluido en el anemómetro sino cable con un **diámetro menor**, puesto que, al utilizar un optoacoplador, la corriente que circula por el cable no supera los **2,5 miliamperios**.

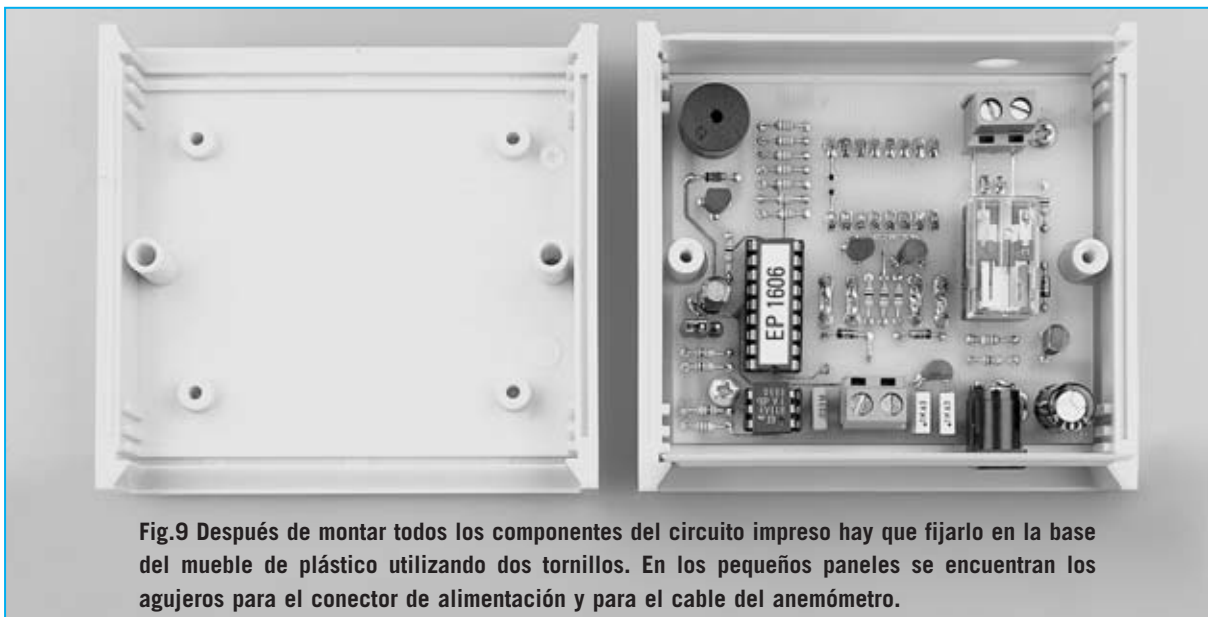




Fig.10 Fotografía del anemómetro y del mueble que contiene el circuito de control. El anemómetro debe ser colocado en una zona donde incida el viento o la corriente de aire que se quiere medir y llevar los cables hasta el circuito de control.

Se puede utilizar cable de **dos hilos** muy **fino** o **cable coaxial**, conectando en este caso la **malla** apantallada a **masa**.

AJUSTE

El pequeño **conector macho** de dos terminales **J1** se utiliza para **ajustar** la **velocidad** que tiene que alcanzar el **viento** para excitar el **relé** y hacer sonar al **zumbador**.

Para programar esta **velocidad** hay que **cortocircuitar** este conector con el pequeño **conector hembra (jumper)** incluido en el kit.

Supongamos que se desea excitar el relé cuando el viento supera los **42 Km/hora**. En primer lugar hay que mantener presionado el **pulsador P1**, automáticamente los números en el display, partiendo de **00**, subirán a **01...02...03...04...** hasta llegar a **39...40...41...42**, momento en el que hay que dejar de presionar el pulsador.

El **pulsador P2** sirve para **decrementar**. Por tanto, si lo presionáramos aparecerían en el display los números **41...40...39...**

Resumiendo:

- El **pulsador P1** sirve para **incrementar** el número.
- El **pulsador P2** sirve para **decrementar** el número.

Una vez ajustado el valor de la **velocidad** del viento a la que se tiene que excitar el **relé** se puede realizar una simulación para comprobar el correcto funcionamiento.

En primer lugar hay que quitar el **conector hembra** de cortocircuito (**jumper**) del **conector J1**, y, a continuación, presionar al mismo tiempo los **pulsadores P1** y **P2**. Automáticamente el valor numérico mostrado en el display comenzará a **incrementarse**, cuando supere el valor de los **42 Km/hora** el **relé** se **excitará**, condición que será confirmada por el encendido del diodo LED **DL1** y por el sonido del **zumbador (buzzer)**.

Si la intensidad sonora del zumbador os parece demasiado **débil** se puede conectar a la clema de salida del **relé** una **sirena** de **alarma**.

PRECIO DE REALIZACIÓN

LX.1606: Precio de todos los componentes necesarios para realizar el **anemómetro programable** mostrado en las Figs.6-8, incluyendo el **conector macho** de alimentación, el **mueble de plástico** y el **sensor anemométrico** mostrado en la Fig.1**113,20 €**

LX.1606: Circuito impreso.....**6,35 €**

SE1.20: Sensor anemométrico**52,65 €**

ESTOS PRECIOS NO INCLUYEN I.V.A.



Se puede utilizar cable de **dos hilos** muy **fino** o **cable coaxial**, conectando en este caso la **malla** apantallada a **masa**.

AJUSTE

El pequeño **conector macho** de dos terminales **J1** se utiliza para **ajustar** la **velocidad** que tiene que alcanzar el **viento** para excitar el **relé** y hacer sonar al **zumbador**.

Para programar esta **velocidad** hay que **cortocircuitar** este conector con el pequeño **conector hembra (jumper)** incluido en el kit.

Supongamos que se desea excitar el relé cuando el viento supera los **42 Km/hora**. En primer lugar hay que mantener presionado el **pulsador P1**, automáticamente los números en el display, partiendo de **00**, subirán a **01...02...03...04...** hasta llegar a **39...40...41...42**, momento en el que hay que dejar de presionar el pulsador.

El **pulsador P2** sirve para **decrementar**. Por tanto, si lo presionáramos aparecerían en el display los números **41...40...39...**

Resumiendo:

- El **pulsador P1** sirve para **incrementar** el número.
- El **pulsador P2** sirve para **decrementar** el número.

Una vez ajustado el valor de la **velocidad** del viento a la que se tiene que excitar el **relé** se puede realizar una simulación para comprobar el correcto funcionamiento.

En primer lugar hay que quitar el **conector hembra** de cortocircuito (**jumper**) del **conector J1**, y, a continuación, presionar al mismo tiempo los **pulsadores P1** y **P2**. Automáticamente el valor numérico mostrado en el display comenzará a **incrementarse**, cuando supere el valor de los **42 Km/hora** el **relé** se **excitará**, condición que será confirmada por el encendido del diodo LED **DL1** y por el sonido del **zumbador (buzzer)**.

Si la intensidad sonora del zumbador os parece demasiado **débil** se puede conectar a la clema de salida del **relé** una **sirena** de **alarma**.

PRECIO DE REALIZACIÓN

LX.1606: Precio de todos los componentes necesarios para realizar el **anemómetro programable** mostrado en las Figs.6-8, incluyendo el **conector macho** de alimentación, el **mueble de plástico** y el **sensor anemométrico** mostrado en la Fig.1**113,20 €**

CC.1606: Circuito impreso**6,35 €**

SE1.20: Sensor anemométrico**52,65 €**

ESTOS PRECIOS NO INCLUYEN I.V.A.

FAMILIA	Código	Descripción	PVP	Revista	Mueble	
TELECOMUNICACIONES	LX 1349	Simple TX-FM para la gama 144-146 MHz	46,43€	170	*	
	LX 1489	Transmisor en CW de 12 vatios en 3 MHz	41,60€	207		
	LX 1555	Radiomicrofono de onda Media	45,65€	229	*	
EMISIÓN	LX 1029	VFO válido de 2 a 200 MHz	36,36€	95		
	LX 1385	VFO programable modulado FM 26-160 MHz	143,46€	182	*	
	LX 1447-48	Timbre portátil red eléct.Emisor/receptor	27,02€	193	Incluido	
	LX 1462	Activador para transmitir en SSB	86,13€	200	*	
	LX 1463	Final RF de 1 vatio	22,84€	199		
	LX 1464	Oscilador para SSB	11,66€	199		
	LX 1490	Microtransmisor FM en 170-173 MHz	112,70€	209	*	
	LX 1557	Transmisor Audio/Vídeo a 2,4 GHz de 20 milivatios	103,70€	232	Incluido	
	ANT.24.8	Antena emisora/receptora para banda 2,4 GHz	96,55€	232		
	LX 1565	VFO programable de 50 180MHz con micro ST7	97,65€	233	incluido	
	LX.1566	Etapa VCO de 100 mW de potencia	60,50€	233		
	LX 5039	Superheterodino para onda media	63,29€	193	*	
	KM 1507	Emisor radiomicrofono FM en 423 MHz	46,90€	214	*	
	LX 1413	Modulador VHF para TV sin Euroconector	29,54€	184	incluido	
	EMISIÓN T.V	KM 1445	Transmitir en 49 canales TV en gama UHF	131,77€	196	
LX 010		Emisora de FM de 1 vatio	40,05€	72-144		
EMISIÓN F.M.	LX 5036	Radiomicrofono FM Banda 88-108 MHz	15,24€	189		
	LX 5037	Sonda de carga para LX 5036	3,43€	189		
EMISIÓN C.B.	LX 5040	Transmisor 27 MHz modulado en AM	33,78€	196		
	LX 5041	Transmisor 27 MHz modulado AM Modulador	26,17€	196		
	LX 5042	Transm.27 MHz mod, AM sonda de carga	4,33€	196		
EMISIÓN COMPLEMENTOS	LX 1248	Codificador estéreo	96,01€	145		
	LX 662	Mini receptor FM	32,45€	23		
RECEPCIÓN	LX 887	Superheterodino didáctico para OM	58,90	64		
	LX 1295	Receptor AM-FM para la gama 110-180 mHz	130,81€	157	*	
	LX 1346	Receptor AM-FM de 38 MHz a 860 MHz	256,66€	171	*	
	KM1450	Módulo SMD para LX. 1451	29,54€	195	*	
	LX 1451	Sintonizador para onda media y FM estéreo	78,52€	195		
	LX 1452	Etapa display para LX 1451	57,40€	195		
	LX 1453	Circuito de ajuste para LX 1451	12,68€	195		
	LX 1519	Recibir onda media con dos integrados	35,10€	217	incluido	
	LX 1529	Receptor FM con solo 3 integrados	51,80€	221		
	LX 1558-58/B	Receptor para la banda de 2,4 GHz	198,70€	232	incluido	
	KM 1508	Receptor Radiomicrofono en FM 423 MHz	83,40€	214	*	
	RECEP.O/CORTA O/LARGA RECEP.COMPLEMENTOS	LX 1532	Redescubrir la fascinante Onda Corta	57,95€		
		LX 1467	E.Alimentación + conmutación para KM1466	46,43€	199	
		KM 1466	Preamplificador de antena de 20 a 450 MHz	5,49€	199	
	SATELITES METEREOLÓGICOS		Parábola rejilla con antena para METEOSAT	164,98€	119	
		ANTENA para satélites polares (doble V)	64,91€	116		
		PREAMPLIFICADOR satélites polares	37,56€	116		
LX 1148		Interface DSP para JVFX	168,88€	125	*	
LX 1375		Receptor para Meteosat y polares	337,53€	180	incluido	
TV.970		Convertor de frecuencia para meteosat	158,22€	180		
LABORATORIO FRECUENCIMETROS	LX 1374	Frecuencímetro digital que lee hasta 2 GHz	167,08€	177	*	
	LX 1374/D	Placa premontada de SMD para LX 1374	29,54€	177		
	LX 1525	Frecuencímetro de 550 MHz con LCD	73,70€	219	incluido	
	LX 1526	Fuente de alimentación LX.1525	23,70€	219		
	LX 1572	Frecuencímetro de 2,2 GHz con 10 dígitos	121,85€	236	incluido	
	LX 5047	Medidor de frecuencia analógico	44,72€	204	incluido	
	LX 5048	Medidor de frecuencia digital de 5 dígitos	139,25€	203	incluido	
	LX 1142	Generador de ruido 1MHz.-2GHz.	79,93€	122	*	
	LX 1234	Generador de VFO sintetizado 1,2 GHz	69,63€	142	*	
	LX 1234/B	Etapa de conmutación completa LX 1234	89,40€	142		
LABORATORIO GENERADORES	LX 1235	Módulos para LX 1234	24,04€	142		
	LX 1344	Etapa de comando	124,89€	170	*	
	LX 1345	Etapa base	168,76€	170		
	LX 1464	Oscilador para SSB	11,66€	199		
	LX 1542	Generador BF con tres formas de ondas	86,10€	222	*	
	LX 1543	Frecuencímetro digital	62,30	222		
	LX1563	Generador de señal RF 40 KHz -13,5 MHz	60,50	233	incluido	
	LX 1151	Generador de BF	31,07€	124	*	
	LX 1337	Generador de BF	56,56€	166	*	
	LX 1513	Generador Sweep B.F.	91,30€	214	*	
LABORATORIO GENERADOR BF	LX 5031	Generador de señal BF	39,67€	178	incluido	
	LX 5032	Generador de señal BF	55,71€	178	incluido	
	LX 1351	Gen.de monoscopio TV/MONITOR VGA	126,57€	171		
	LX 1125	Medidor flujo magnético	56,04€	119		
	LX 1192	Impedancímetro y Reactancímetro	179,31€	134	*	
LAB.GENERADOR TV LABORATORIO MEDIDORES	LX 1310	Medidor de campos electromagnéticos	84,44€	159	Incluido	
	LX 1393	Para medir imped. característica de antena	25,33€	185		
	LX 1421	Localizador de terminales de un transistor	46,85€	187	incluido	

FAMILIA	Código	Descripción	PVP	Revista	Mueble
	LX 1431	Analizador RF para osciloscopio	105,48€	192	*
	LX 1432	Fuente de Alimentación para LX 1431	37,98€	192	
	LX 1435- /B	Contaminación e. irradiada por enlaces RF	115,60€	193	
	LX 1512	Medidor de Tierra	66,20€	215	*
	LX 1518	Medir la ESR de un condensador electrolítico	36,85€	216	
	LX 1522	Como controlar el valor de una inductancia	38,60€	216	
	LX 1538	Trazador de curvas para Transistores-Fet,SCR etc.	122,85€	224	*
	LX 1556	Voltímetro-Amperímetro digital	74,30€	232	*
	LX 1570	Termómetro a distancia	126,15€	235	incluido
	LX 1576	Inductancímetro de 0,1 a 300 microHenrios	60,50€	237	
LAB. COMPROBADORES	LX 1272	Comprobador de Mospower Mosfet e IGBT	23,65€	152	
	LX 5014	Comprobador de transistores	61,60€	160	incluido
LAB. COMPLEMENTOS	LX 5019	Comprobador para SCR y TRIAC	72,15€	166	incluido
	LX 1169	Preamplificador 400 KHz.- 2GHz.	27,05€	128	
	LX 1456	Preamplificador de antena de 0,4 a 50 MHz	18,18€	197	
SONIDO HI-FI	LX 1113	Ampl. HI-FI estéreo con válvulas. EL34	325,63€	115	*
SONIDO AMPLIFICADORES		Ampl. HI-Fi estéreo con válvulas KT88	371,43€	115	
	LX 1114	Fuente de alimentación para LX 1113	142,08€	115	
	LX 1115	Vú-meter para amplificadores	18,00€	115	
	LX 1239	Fuente de alimentación para LX 1240	56,28€	142	
	LX 1240	Amplificador estéreo para EL 34	159,00€	142	*
	LX 1257	Fuente de alimentación para LX 1256	69,72€	148	
	LX 1258	V-Meter para LX 1256	39,85€	148	
	LX 1309	Amplificador a válvulas para auriculares	139,25€	160	*
	LX 1320	Amplificador compacto a válvulas	171,89€	161	*
	LX 1321	Etapa final para LX 1320	421,91€	161	
	LX 1322	Etapa Vu-meter para LX 1320	62,51€	161	
	LX 1323	Fuente de alimentación para LX 1320	179,70€	161	
	LX 1471	Final estéreo Hi-Fi de 110+110 vatios musicales	75,25€	211	incluido
	LX 1472	Amplificador HI-FI de 200 W con finales IGBT	66,25€	213	*
	LX 1473	Final con mospower de 38-70 vatios RMS	44,20€	212	*
	LX 1553	Amplificador SUB-WOOFER con filtro DIGITAL	171,10€	231	*
	LX 1577	Amplificador HI-FI 30 vatios RMS sobre 8 Ohmios	39,75€	236	*
	LX 1578	Etapa de alimentación para LX.1577	51,55€	236	
	LX 5043	Convertir la gama de 27 MHz en onda media	26,17€	197	
SONIDO HI-FI PREVIOS	LX 1139	Etapa entrada LX 1140	46,28€	122	
	LX 1140	Previo estéreo a válvulas	214,26€	122	*
	LX 1141	Etapa alimentación LX 1140	82,94€	122	
	LX 1149	Previo Hi-Fi a Fet	63,23€	125	
	LX 1150	Previo Hi-Fi a Fet	53,88€	125	*
SONIDO HI-FI COMPLEM.	LX 1169	Amplificador de 400 khz a 2 GHz	27,05€	128	
	LX 1073	Filtro estéreo paso alto	24,04€	104	
	LX 1074	Filtro estéreo paso bajo	23,14€	104	
	LX 1198-/B	Filtro cross-over estéreo	71,73€	135	*
	LX 1241	Mezclador a fet	58,45€	144	*
	LX 1242	Mezclador a fet (00es)	44,78€	144	
	LX 1275	Micrófono para escuchar a distancia	40,51€	154	
	LX 1282	Compresor ALC estéreo	98,75€	153	
	LX 1357	Ecuilizador RIAA con filtro antiruido	36,30€	174	
	LX 1564	Karaoke con efecto eco	63,10€	234	*
FUENTES DE ALIMENTACION	LX 1131	Fuente de Alimentación 3-18 V 2A.	27,05€	121	
	LX 1138	Cargador de baterías plomo	84,74€	122	
	LX 1364	Al. de 2,5 a 25 V. max.5 amp. Etapa base	61,90€	175	*
	LX 1364/B	Al. de 2,5 a 25 V. max.5 amp. Etapa final	16,50€	175	
	LX 1364/C	Al.de 2,5 a 25 V. max.5 amp.Etapa voltímetro	39,88€	175	
	LX 1449	Inversor de 12 volt. CC a 220 volt. AC 50 Hz	202,54€	197	*
CARGADORES	LX 1545	Alimentador estabilizado	78,95€	226	*
	LX 1069	Cargador de baterías de niquel-cadmio	64,91€	103	*
	LX 1428	Cargador bat. automáticos con diodos SCR	121,07€	190	
	LX 1479	Cargador de pilas NI-MH	109,71€	201	*
SEGURIDAD ALARMAS	LX 1396	RADAR antirrobo de 10 gHz	50,49€	184	incluido
	LX 1424	Antirrobo banda UHF 433,9 MHz transmisión	56,98€	190	incluido
	LX 1425	Antirrobo banda UHF 433,9 MHz recepción	60,76€	190	incluido
	LX 1506	Alarma por sensor volumétrico	40,40€	209	*
SEGURIDAD SIRENAS SEG. COMPLEMENTOS	LX 5025	Sirena bitonal digital	19,41€	170	
	LX 5027	Contador 2 cifras	27,86€	172	
	LX 5028	Contador 2 cifras	25,33€	172	
SEGURIDAD DETECTORES	LX 1216	Detector para fugas de gas	77,74€	137	
	LX 1287	Detector para micrófonos	35,46€	155	
	LX 1407	Nuevo y eficaz contador geiger	139,25€	185	incluido
	LX 1433	Buscador de cables instalaciones eléctricas	16,47€	192	incluido
	LX 1465	Sensible detector de metales	88,60€	216	*
	LX 1517	Detector de fugas para Micro-ondas	34,75€	217	incluido
	LX 1568	Emisor de Barrera de Rayos infrarrojos	10,40€	234	incluido
	LX 1569	Receptor de Barrera de Rayos infrarrojos	20,75€	234	incluido

FAMILIA	Código	Descripción	PVP	Revista	Mueble
MEDICINA ELECTRÓNICA	LX 559	Detector de acupuntura	17,13€	8	
	LX 654	Acupuntura portátil	23,14€	24	
	LX 811	Electromagnetoterapia reforzada en A.F.	66,71€	55/147	*
	LX 811/B	Disco radiante para LX 811	12,32€	55	
	LX 950	Electromagnetoterapia en baja frecuencia	49,58€	77	*
	LX 950/B	Difusor para LX 950	10,82	77	
	MP 950	Difusor magnético	10,82€	77	
	LX 987	Etapa de potencia para LX 950	21,34€	85	
	LX 1003	Estimulador analgésico	41,47€	90	
	LX 1010	Iones negativos para coche	39,07€	90	
	LX 1072	Banda radiante para LX 811	15,93€	104	
	LX 1146	Magnetoterapia BF alta eficacia	212,01€	123	incluido
	MP 90	Difusor magnético	28,25€	123	
	LX 1176	Cargador de baterías para LX 1175	37,83€	129	
	LX 1293	Magnetoterapia de AF	156,11€	157	incluido
	PC 1293	Paño radiante para LX.1293	37,98€	157	
	LX 1343	Depurador antipolución	101,27€	169	incluido
	LX 1365	Nueva lontoforesis con microprocesador	75,97€	175	*mo 1365
	LX 1365/B	Circuito display	24,91€	175	
	LX 1365/P	Placa de aplicación	16,47	175	
	LX 1387	Tens, electromedicamento elimina el dolor	84,74€	181	*
	LX 1387/B	Placa de visualización	40,93€	181	
	LX 1408	Tonificar los músculos con la electrónica	118,16€	186	
LX 1480	Ionoterapia	106,38€	202	incluido	
LX 1480-B	Etapa Voltímetro para LX.1480	36,66€	202		
LUCES-ILUMINACIÓN	LX 1011	Generador de albas y ocasos digital 1 salida	61,90€	91	
	LX 1061	Luces tremolantes	50,49€	107	
	LX 1326	Luz que apaga y se enciende gradualmente	47,69€	165	*
	LX 1493	Generador de Alba y ocaso	101,27€	206	incluido
	MISCELANEA	LX 1025	Termostato con relé	44,47€	96
LX 1182		Temporizador variable	46,43€	130	
LX 1238		Circuito simulador de rayos	35,79€	143	
LX 1259		Ahuyentador de mosquitos	44,75€	151	Incluido
LX 1332		Ahuyenta-ratones ultrasónico	39,25€	167	*
LX 1398		Vallas con descargas de Electroshock	27,02€	186	
LX 1562		Alimentador PWM para TRENES ELECTRICOS	112,35€	232	*
LX 5035		Reloj digital	84,44€	185	*
LX 5044		Temporizador con el NE.555	24,07€	198	*
LX 5045		Temporizador con el NE.555	26,17€	198	
CIRCUITOS DIDÁCTICOS	LX 1325	Programador para MICRO ST6 60/65	84,44€	165	*
	LX 1329	Entrenador para ST6/60-65	32,09€	166	
	LX 1329/B	Interface para ST6/60-65	14,36€	166	
	LX 1546	Programador para ST7-lite 09	26,65€	227	
	LX 1547	Entrenador para LX.1546	53,60€	227	
	LX 1548	Tarjeta experimental reloj para ST7	23,70€	228	
	LX1549	Tarjeta experimental display para ST7	36,05€	228	
CIRCUITOS TELÉFONO	LX 1510	Excitar un relé con un teléfono	109,10€	213	*
	KM 1515	Leer y escribir en las tarjetas sim de los móviles	78,95€	216	
MANDO A DISTANCIA	LX 1409	Telemando codificado de 4 canales Transmisor	24,49	184	incluido
	LX 1410	Telemando codificado de 4 canales Receptor	58,24	184	*
	LX 1411	Salida de 2 relés para el LX.1410	21,94	184	
	LX 1412	Salida de 4 relés para el LX.1410	32,06	184	
	LX 1474	Mando a distancia a 433 MHz via radio -Transmisor	63,80	199	incluido
	LX 1475	Mando a distancia a 433 MHz via radio - Receptor	84,44	199	incluido
	LX 1501	Mando Emisor codificado a traves de red eléctrica	58,15€	210	incluido
	LX 1502	Receptor de LX1501	64,65€	210	incluido
ORDENADORES	LX 1574	Programador de EPROM para puerto paralelo	82,95€	237	
	LX 1575	Etapa de soporte para LX 1574	31,10€	237	

¡MAS DE 800 MONTAJES DISPONIBLES! www.nuevaelectronica.com

Nº238 - ESTOS PRECIOS NO INCLUYEN I.V.A. *Esta lista anula las anteriores.* * consultar precio del mueble 91 542 73 80

Casi todo el mundo sabe que los murciélagos utilizan un radar ultrasónico para identificar objetos en la oscuridad y también que los perros son capaces de captar los ultrasonidos. ¿Nunca os habéis preguntado si un ser humano puede oírlos? La respuesta a esta pregunta la ofrece este proyecto.



FUNCIONAMIENTO Y ESQUEMA ELÉCTRICO

La membrana del tímpano de un **oído humano** es capaz de vibrar desde un mínimo de **20 Hz** hasta un máximo de **20.000 Hz (20 KHz)**. Las frecuencias superiores a este valor **no** las podemos oír ya que nuestro sistema auditivo está incapacitado para ello, lo cual no quiere decir que no existan sonidos a **más frecuencia** de 20.000 Hz (denominados **ultrasonidos**). De hecho animales tan comunes como los **perros** o los **murciélagos** son capaces de escuchar frecuencias ultrasónicas.

Para superar nuestra **incapacidad** de captación de ultrasonidos hemos desarrollado un circuito capaz de captar ultrasonidos y **transformarlos** en sonidos **audibles** utilizando una **cápsula microfónica ultrasónica** que capta frecuencias entre **20 KHz y 100 KHz**. Las frecuencias superiores a 20 KHz, inaudibles para el oído humano, son transformadas a frecuencias entre **300 Hz y 10 KHz**.

Observando el esquema eléctrico se puede apreciar que las señales captadas por el **micrófono ultrasónico (MIC)** son amplificadas a través del FET **FT1** y del integrado **IC1**. El valor de la amplificación es de unas **100 veces** en tensión, valor que corresponde a **40 dB**.

La señal amplificada se aplica a la patilla de entrada (1) de **IC3**, un **mezclador/conversor** tipo **NE.602**. A la patilla 6 de este integrado se aplica una señal de **frecuencia**

variable (20 KHz a 150 KHz) generada por un integrado tipo **CD.4046 (IC2)**.

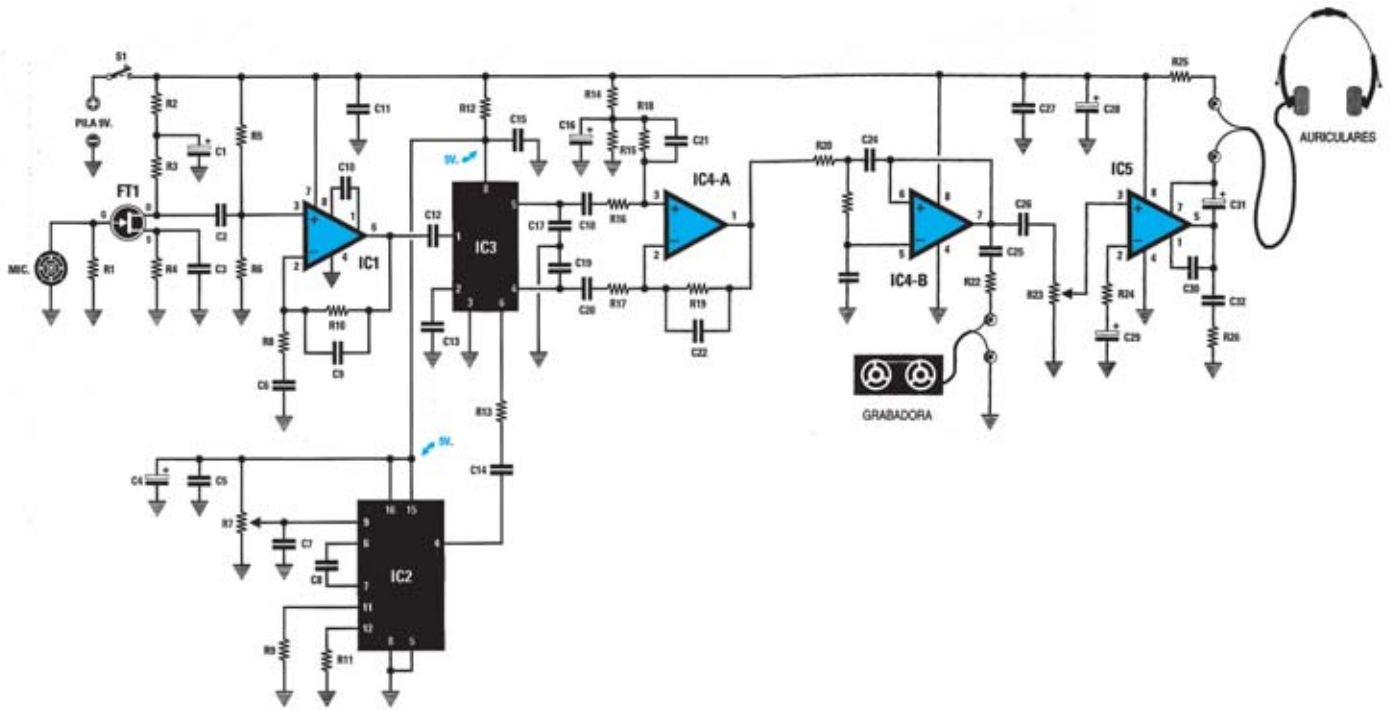
Girando el mando del **potenciómetro R7** para que se generen diferentes valores de **frecuencia**, la señal captada reducirá su frecuencia en el valor fijado haciendo que la alta frecuencia de los ultrasonidos sea **audible**. Todas las frecuencias audibles obtenidas por esta conversión se obtienen de las patillas de salida (4-5) de **IC3** y se aplican a la entrada del operacional **IC4/A**.

De la patilla de salida (1) de **IC4/A** la señal se aplica a la entrada de **IC4/B**, un operacional utilizado como **filtro paso-bajo** con una frecuencia de **corte de 10.000 Hz**. Por tanto, a su salida están disponibles todas las frecuencias ultrasónicas convertidas en frecuencias **audibles** no superiores a **10 KHz**.

Para **grabar** las señales captadas se puede conectar una grabadora directamente a la toma de la resistencia **R22**, mientras que para escucharlas a través de **auriculares** se amplifican a través de un **TBA.820/M (IC5)**.

El **potenciómetro R23**, situado entre la salida de **IC4/B** y la entrada de **IC5**, sirve para **ajustar** la **amplitud** de la señal captada.

El circuito se **alimenta** a través de una pila corriente de **9 voltios**.



Esquema eléctrico y lista de componentes del Detector de Ultrasonidos LX.1226.

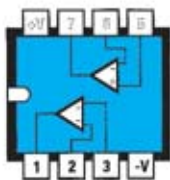
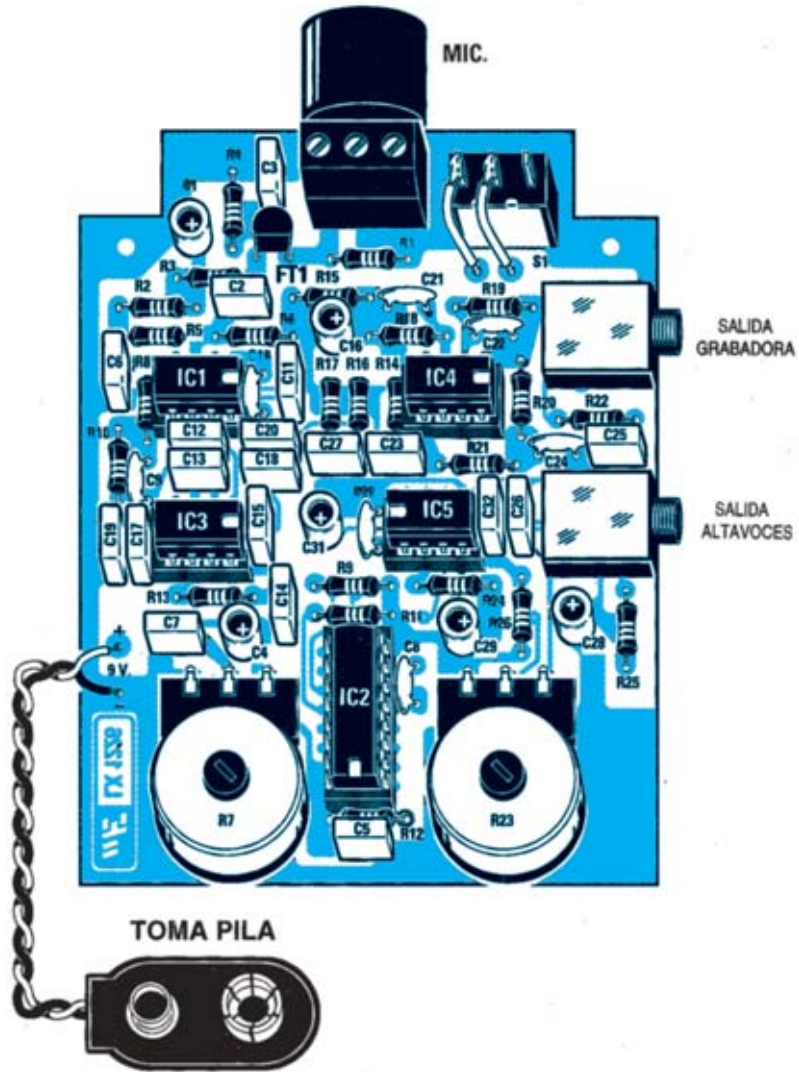
**LISTA DE COMPONENTES
LX.1226**

R1 = 1 megaohm 1/4 wat
 R2 = 100 ohm 1/4 wat
 R3 = 3.900 ohm 1/4 wat
 R4 = 1.000 ohm 1/4 wat
 R5 = 100.000 ohm 1/4 wat
 R6 = 100.000 ohm 1/4 wat
 R7 = 10.000 ohm pot. lin.
 R8 = 1.000 ohm 1/4 wat
 R9 = 10.000 ohm 1/4 wat
 R10 = 33.000 ohm 1/4 wat
 R11 = 150.000 ohm 1/4 wat
 R12 = 470 ohm 1/4 wat
 R13 = 100.000 ohm 1/4 wat
 R14 = 6.800 ohm 1/4 wat
 R15 = 6.800 ohm 1/4 wat
 R16 = 4.700 ohm 1/4 wat
 R17 = 4.700 ohm 1/4 wat
 R18 = 68.000 ohm 1/4 wat
 R19 = 68.000 ohm 1/4 wat
 R20 = 22.000 ohm 1/4 wat
 R21 = 22.000 ohm 1/4 wat

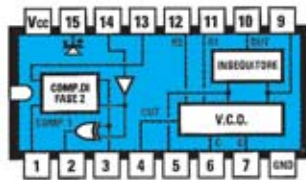
R22 = 820 ohm 1/4 wat
 R23 = 47.000 ohm pot. log.
 R24 = 100 ohm 1/4 wat
 R25 = 39 ohm 1/4 wat
 R26 = 1 ohm 1/4 wat
 C1 = 10 mF electr. 63 volt
 C2 = 1.000 pF poliester
 C3 = 47.000 pF poliester
 C4 = 10 mF electr. 63 volt
 C5 = 100.000 pF poliester
 C6 = 10.000 pF poliester
 C7 = 47.000 pF poliester
 C8 = 470 pF cerámico
 C9 = 15 pF cerámico
 C10 = 56 pF cerámico
 C11 = 100.000 pF poliester
 C12 = 10.000 pF poliester
 C13 = 10.000 pF poliester
 C14 = 10.000 pF poliester
 C15 = 100.000 pF poliester
 C16 = 10 mF electr 63 volt
 C17 = 10.000 pF poliester
 C18 = 100.000 pF poliester

C19 = 10.000 pF poliester
 C20 = 100.000 pF poliester
 C21 = 220 pF cerámico
 C22 = 220 pF cerámico
 C23 = 1.000 pF poliester
 C24 = 560 pF cerámico
 C25 = 220.000 pF poliester
 C26 = 100.000 pF poliester
 C27 = 100.000 pF poliester
 C28 = 100 mF electr. 25 volt
 C29 = 100 mF electr. 25 volt
 C30 = 340 pF cerámico
 C31 = 100 mF electr. 25 volt
 C32 = 220.000 pF poliester
 FT1 = fet tipo MPF102
 IC1 = CA.3130
 IC2 = CMOS 4046
 IC3 = NE.602
 IC4 = TL.082
 IC5 = TBA.820M
 MIC= micrófono ultrasónico
 S1 = interruptor
 AURICULARES 32 ohm

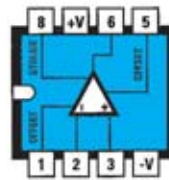
MONTAJE Y AJUSTE



TL 082



4046



CA 3130



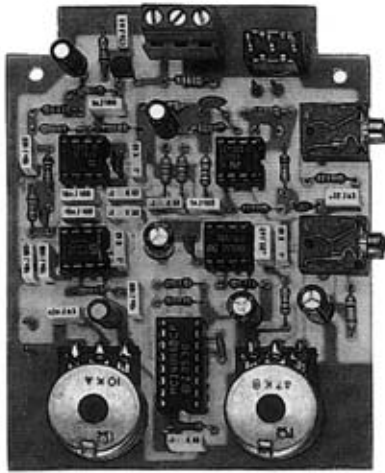
TBA 820M



NE 602

Esquema de montaje práctico de la placa LX.1226 y disposición de terminales de los semiconductores utilizados en el circuito.





Aspecto final del circuito LX.1226 y montaje en el mueble, no incluido en el kit (hay que solicitarlo aparte).



Para realizar el Detector de Ultrasonidos se necesita un **circuito impreso** de doble cara: El **LX.1226**, circuito que soporta todos los componentes. Para el montaje es importante tener presentes las siguientes consideraciones.

Zócalos: Al montar los **zócalos** para los circuitos integrados **IC1, IC2, IC3, IC4 e IC5** hay que respetar la muesca de referencia presente en la serigrafía del circuito impreso y no utilizar mucho estaño para no provocar cortocircuitos.

Resistencias: Cuando se monten las **resistencias** que incluye el circuito (**R1-R6, R8-R22, R24-R26**) hay que controlar su valor óhmico, si es preciso con la ayuda de una tabla de colores. En el caso de los **potenciómetros** (**R7, R23**), que se han de soldar directamente al circuito impreso como se muestra en el esquema de montaje práctico, el valor se controla mediante la serigrafía impresa sobre su cuerpo.

Condensadores: Hay que controlar su valor por la serigrafía impresa en su cuerpo. Al montar los de **poliéster** (**C2-C3, C5-C7, C11-C15, C17-C20, C23, C25-C27, C32**) y los **cerámicos** (**C8-C10, C21-C22, C24, C30**) no hay que preocuparse por la polaridad ya que carecen de ella. En cambio, al montar los condensadores **electrolíticos** (**C1, C4, C16, C28-C29, C31**) sí hay que tener en cuenta la polaridad de sus terminales.

Semiconductores: Al realizar el montaje del **FET (FT1)** hay que soldarlo respetando la disposición de terminales, para lo cual hay que orientar su lado plano tal y como se indica en el esquema de montaje práctico.

Conectores: Este circuito incluye una **clema de 3 polos** para la conexión del **micrófono ultrasónico** y **2 conectores hembra** tipo **Jack** utilizados para la conexión de **auriculares** y de una **grabadora**. También

incluye un **portapilas de 9 voltios** cuyos cables de conexión se sueldan directamente al impreso, teniendo cuidado en respetar su polaridad (cable rojo al positivo y cable negro al negativo).

Interruptores y pulsadores: El **interruptor** de encendido (**S1**) se ha de fijar directamente en el circuito impreso, tal y como se muestra en el esquema de montaje práctico.

Circuitos integrados con zócalo: Los integrados **IC1, IC2, IC3, IC4 e IC5** se han de introducir en sus correspondientes zócalos haciendo coincidir las muescas de referencia en forma de **U** de los integrados con la de los zócalos.

Elementos diversos: El kit incluye un **micrófono ultrasónico (MIC)** que se instala conectándolo a la clema de tres polos, por tanto sin realizar ninguna soldadura.

MONTAJE EN EL MUEBLE: En el kit **no** incluye mueble contenedor, si se desea se puede pedir el contenedor de plástico **MO.1226** que incluye un **panel serigrafiado**. Antes de fijar el circuito impreso dentro del mueble hay que realizar en el mueble los **orificios** para los potenciómetros, el interruptor y los conectores.

AJUSTE Y PRUEBA: Este circuito no precisa ningún ajuste.

UTILIZACIÓN: Detectar ultrasonidos es muy sencillo con este Detector, de hecho basta con dirigir la cápsula microfónica hacia **cualquier dirección** y luego girar el potenciómetro de ajuste de la **Frecuencia (R7)** hasta percibir algún sonido. Como prueba se puede hacer mover una pulsera metálica, aunque aparentemente no se produce ningún ruido, con los auriculares se escuchará un ruido ensordecedor.

PRECIOS Y REFERENCIAS

LX.1226: Todos los componentes necesarios para la realización del Detector de Ultrasonidos, únicamente excluido el mueble contenedor	60,10 € + IVA
MO.1226: Mueble de plástico con panel serigrafiado.....	19,65 € + IVA
LX.1226: Circuito impreso.....	15,63 € +IVA

Revista de aparición del kit: **N.141**

Conectado la salida de este Codificador FM Estéreo (Stereo FM Encoder) a la entrada de cualquier Transmisor FM Mono pasará a transmitir en estéreo la señal estereofónica presente en la entrada del Codificador.



FUNCIONAMIENTO Y ESQUEMA ELÉCTRICO

Cuando sintonizamos una **emisora FM** que transmite en **estéreo** oiremos por nuestro altavoz derecho la señal captada por el micrófono derecho de la emisora y por nuestro altavoz izquierdo la señal captada por el micrófono izquierdo de la emisora. Para transmitir y recibir **dos señales diferentes** en una única onda, y además compatibilizarlo con dispositivos mono, se utiliza un **Codificador FM Estéreo** en la **emisora** y un **Decodificador FM Estéreo** en los **receptores**.

Este sistema de codificación-decodificación esta **estandarizado** a nivel mundial para **compatibilizar** todas las emisoras y los receptores. Se basa en la transmisión en diferentes frecuencias de las señales: En la banda **0-15 KHz** se transmite la señal mezclada de los dos canales (para receptores **mono**), en los **19 KHz** se transmite el **identificador** de señal **Mono/Estéreo**, en la banda **23KHz-38KHz** se transmite la señal el **canal izquierdo** y en la banda **38KHz-53KHz** se transmite la señal el **canal derecho**. El circuito **LX.1248** responde a este estándar de transmisión, por lo que se puede utilizar con **cualquier emisora FM** garantizando que todos los receptores FM Estéreo del mercado recibirán la señal en perfectas condiciones.

Empezamos la descripción del esquema eléctrico por el integrado **IC1**, un **CD.4046**, utilizado para obtener las frecuencias de **38 KHz** y de **19 KHz**. **MF1**, conectada a las patillas **10-11** del oscilador interno de **IC1**, permite obtener en la patilla **9** una frecuencia de **608 KHz**, frecuencia que queda **dividida por 16** en la patilla **7** y **dividida por 32** en la patilla **5**. Por tanto en la patilla **7** hay una frecuencia de

608:16= 38 KHz y en la patilla **5** una frecuencia de **608:32=19 KHz**.

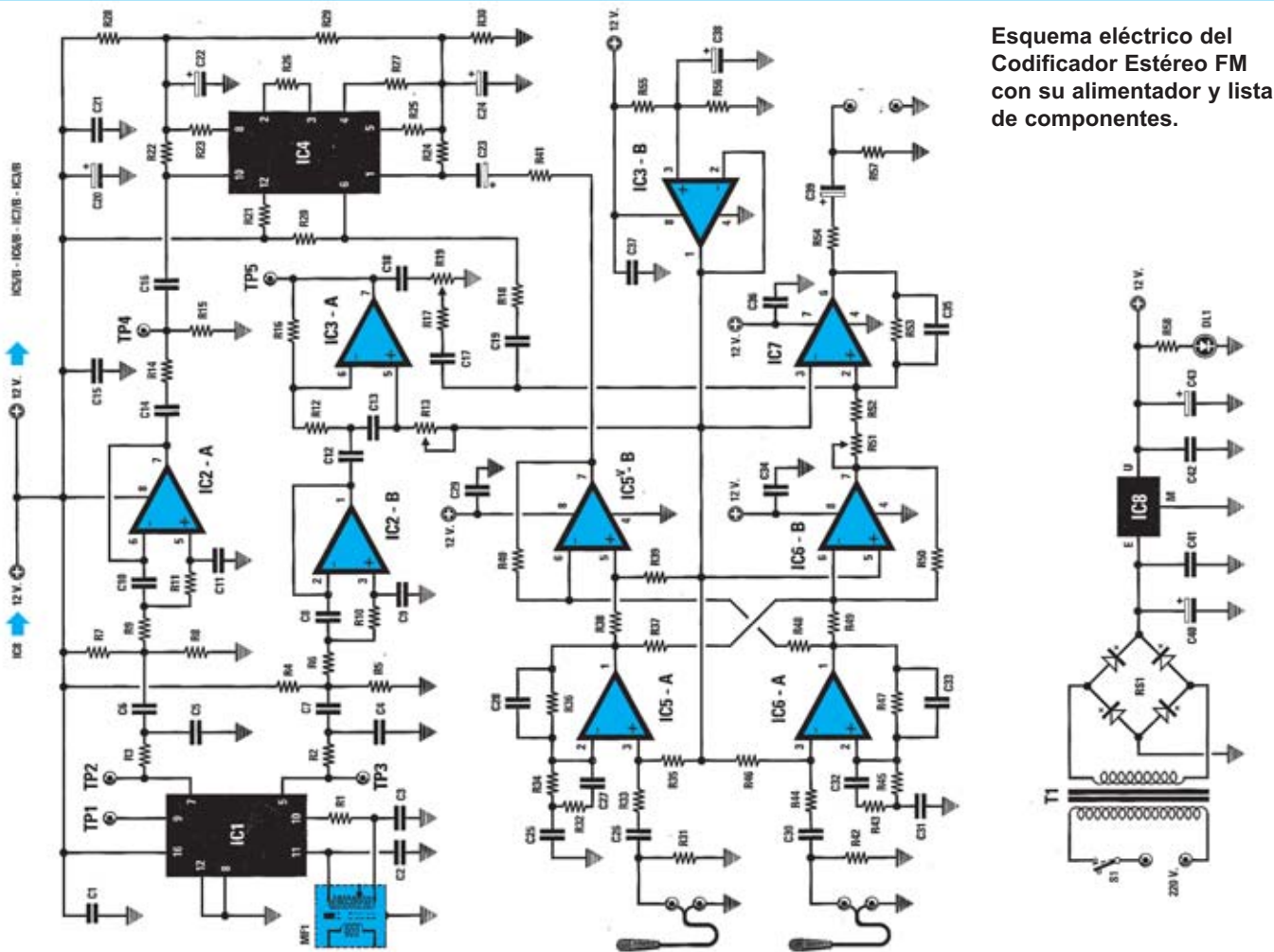
Para que estas señales sean perfectamente sinusoidales a las frecuencias requeridas hemos introducido un **filtro paso-bajo** con una frecuencia de corte de **45 KHz** para la señal de **38 KHz (IC2/A)** y otro **filtro paso-bajo** con una frecuencia de corte de **22 KHz** para la señal de **19 KHz (IC2/B)**. La frecuencia de **38 KHz**, utilizada como **portadora**, se aplica mediante **C14-C16** a la patilla **10** de **IC4**, un **modulador balanceado LM.1496**.

Las señales de las entradas son **preamplificadas** por los operacionales **IC5/A** e **IC6/A**. El operacional **IC6/B suma** la amplitud de la señal del canal izquierdo con la amplitud de la señal de canal derecho (señal que se aplica al operacional **IC7** utilizado como **mezclador**) mientras que **IC5/B resta** la amplitud de la señal del canal izquierdo de la amplitud de la señal de canal derecho (señal que se aplica al **modulador IC4**).

El último operacional que hay en este esquema, denominado **IC3/B**, se utiliza únicamente para obtener un valor de tensión igual a la **mitad de la tensión de alimentación** y así alimentar las patillas **no inversoras** de los operacionales **IC5/A, IC5/B, IC6/A, IC6/B** e **IC3/A**.

El circuito se alimenta con una tensión **estabilizada** de **12 voltios** que se obtiene del integrado **ua.7812 (IC8)** que hay en la etapa de alimentación **incluida** en el propio circuito.





Esquema eléctrico del Codificador Estéreo FM con su alimentador y lista de componentes.

LISTA DE COMPONENTES LX.1248

- | | | | |
|---------------------------|-----------------------------|-------------------------------|--|
| R1 = 4.700 ohm 1/4 wat | R34 = 47.000 ohm 1/4 wat | C9 = 560 pF poliéster | C42 = 100.000 pF poliéster |
| R2 = 1.000 ohm 1/4 wat | R35 = 47.000 ohm 1/4 wat | C10 = 1.000 pF poliéster | C43 = 100 mF elect. 50 volt |
| R3 = 1.000 ohm 1/4 wat | R36 = 100.000 ohm 1/4 wat | C11 = 560 pF cerámico | MF1 = media freq. 455 KHz ROSA |
| R4 = 10.000 ohm 1/4 wat | R37 = 20.000 ohm 1/4 wat 1% | C12 = 47.000 pF poliéster | DI1 = diodo led |
| R5 = 10.000 ohm 1/4 wat | R38 = 20.000 ohm 1/4 wat 1% | C13 = 1.000 pF poliéster | RS1 = puente rectificador 100 volt 1 amper |
| R6 = 10.000 ohm 1/4 wat | R39 = 20.000 ohm 1/4 wat 1% | C14 = 68.000 pF poliéster | IC1 = C/Mos tipo 4060 |
| R7 = 10.000 ohm 1/4 wat | R40 = 20.000 ohm 1/4 wat 1% | C15 = 100.000 pF poliéster | IC2 = MC.4558 |
| R8 = 10.000 ohm 1/4 wat | R41 = 680 ohm 1/4 wat | C16 = 100.000 pF poliéster | IC3 = MC.4558 |
| R9 = 4.700 ohm 1/4 wat | R42 = 47.000 ohm 1/4 wat | C17 = 68.000 pF poliéster | IC4 = LM.1496 ó MC.1496 |
| R10 = 10.000 ohm 1/4 wat | R43 = 8.200 ohm 1/4 wat | C18 = 100.000 pF poliéster | IC5 = MC.4558 |
| R11 = 4.700 ohm 1/4 wat | R44 = 47.000 ohm 1/4 wat | C19 = 4.700 pF poliéster | IC6 = MC.4558 |
| R12 = 22.000 ohm 1/4 wat | R45 = 47.000 ohm 1/4 wat | C20 = 47 mF elect. 25 volt | IC7 = TL.081 |
| R13 = trimmer 100.000 ohm | R46 = 47.000 ohm 1/4 wat | C21 = 100.000 pF poliéster | IC8 = uA.7812 |
| R14 = 10.000 ohm 1/4 wat | R47 = 100.000 ohm 1/4 wat | C22 = 47 mF elect. 25 volt | T1 = trasform. 6 wat (T006.02) |
| R15 = 1.000 ohm 1/4 wat | R48 = 20.000 ohm 1/4 wat | C23 = 47 mF elect. 25 volt | sec. 8/15 volt - 400 mA |
| R16 = 22.000 ohm 1/4 wat | R49 = 20.000 ohm 1/4 wat | C24 = 47 mF elect. 25 volt | S1 = interruptor |
| R17 = 22.000 ohm 1/4 wat | R50 = 20.000 ohm 1/4 wat | C25 = 220.000 pF poliéster | |
| R18 = 5.600 ohm 1/4 wat | R51 = trimmer 50.000 ohm | C26 = 220.000 pF poliéster | |
| R19 = 2.000 ohm trimmer | R52 = 3.300 ohm 1/4 wat | C27 = 1.000 pF poliéster | |
| R20 = 3.900 ohm 1/4 wat | R53 = 22.000 ohm 1/4 wat | C28 = 100 pF cerámico | |
| R21 = 3.900 ohm 1/4 wat | R54 = 100 ohm 1/4 wat | C29 = 100.000 pF poliéster | |
| R22 = 1.000 ohm 1/4 wat | R55 = 47.000 ohm 1/4 wat | C30 = 220.000 pF poliéster | |
| R23 = 1.000 ohm 1/4 wat | R56 = 47.000 ohm 1/4 wat | C31 = 220.000 pF poliéster | |
| R24 = 1.000 ohm 1/4 wat | R57 = 47.000 ohm 1/4 wat | C32 = 1.000 pF poliéster | |
| R25 = 6.800 ohm 1/4 wat | R58 = 8200 ohm 1/4 wat | C33 = 100 pF cerámico | |
| R26 = 1.000 ohm 1/4 wat | C1 = 100.00 pF poliéster | C34 = 100.000 pF poliéster | |
| R27 = 1.000 ohm 1/4 wat | C2 = 470 pF cerámico | C35 = 120 pF cerámico | |
| R28 = 220 ohm 1/4 wat | C3 = 470 pF cerámico | C36 = 100.000 pF poliéster | |
| R29 = 100 ohm 1/4 wat | C4 = 22.000 pF poliéster | C37 = 100.000 pF poliéster | |
| R30 = 220 ohm 1/4 wat | C5 = 10.000 pF cerámico | C38 = 22 mF elect. 25 volt | |
| R31 = 47.000 ohm 1/4 wat | C6 = 47.000 pF poliéster | C39 = 10 mF elect. 63 volt | |
| R32 = 8.200 ohm 1/4 wat | C7 = 47.000 pF poliéster | C40 = 1.000 mF elect. 50 volt | |
| R33 = 47.000 ohm 1/4 wat | C8 = 1.000 pF poliéster | C41 = 100.000 pF poliéster | |

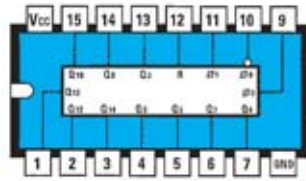
MONTAJE Y AJUSTE



TL 081

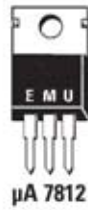


MC 4558



4060

ENTRADA NO INV. C1	0V
REG. GANANCIA C2	N.C.
REG. GANANCIA C3	1/2 SALIDA INV.
ENTRADA INV. C4	11 N.C.
BIAS C5	10 ENTRADA SEÑAL
SALIDA NO INV. C6	9 N.C.
N.C. C7	8 ENTRADA SEÑAL



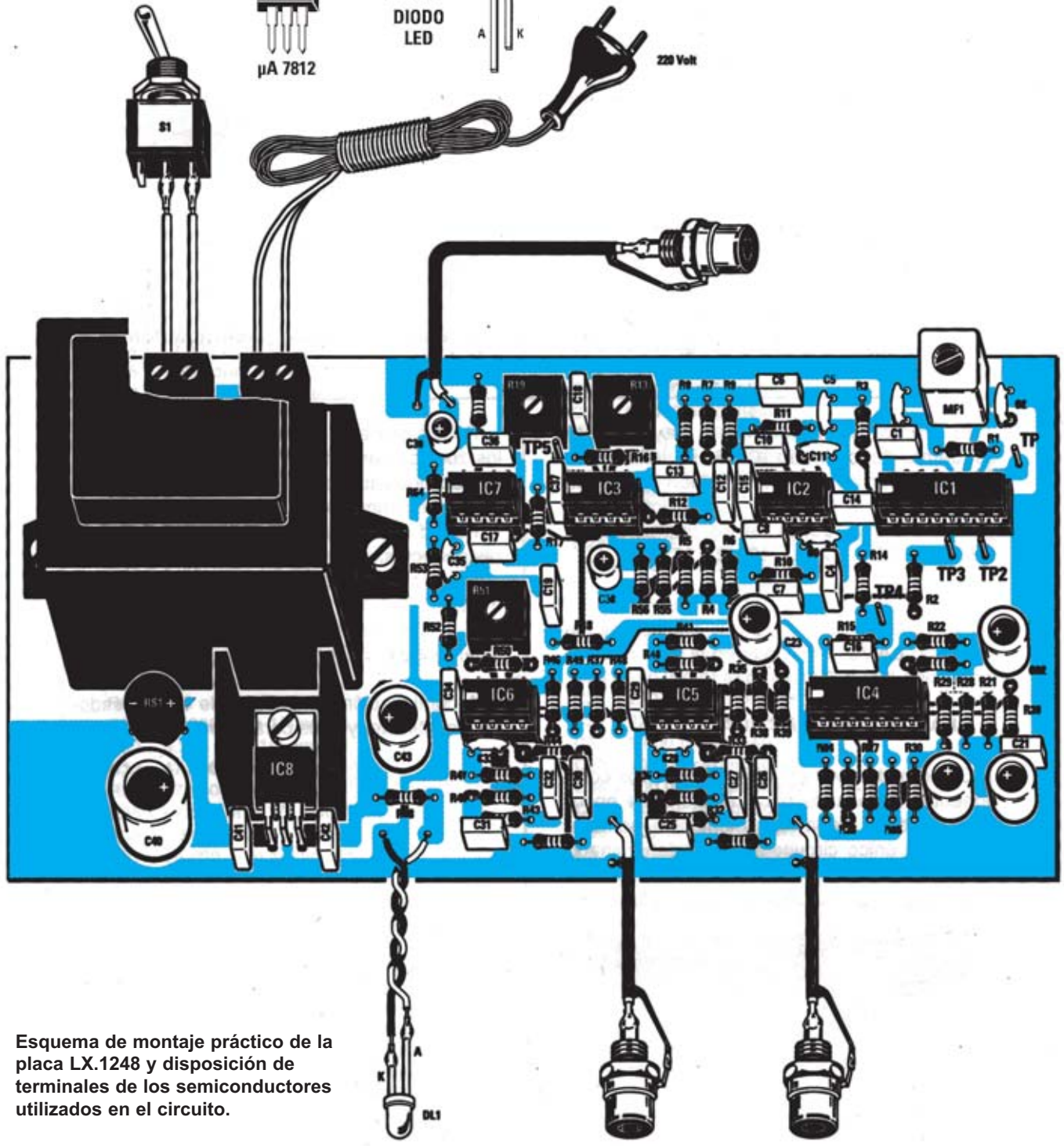
μA 7812



DIODO LED



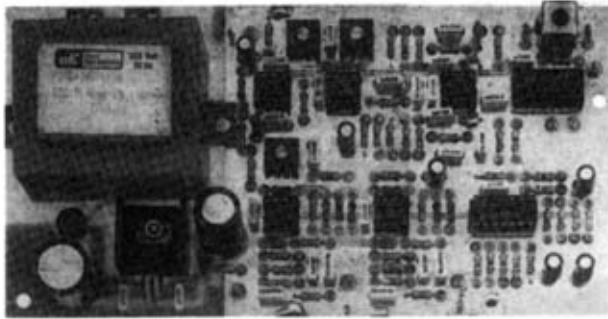
220 Volt



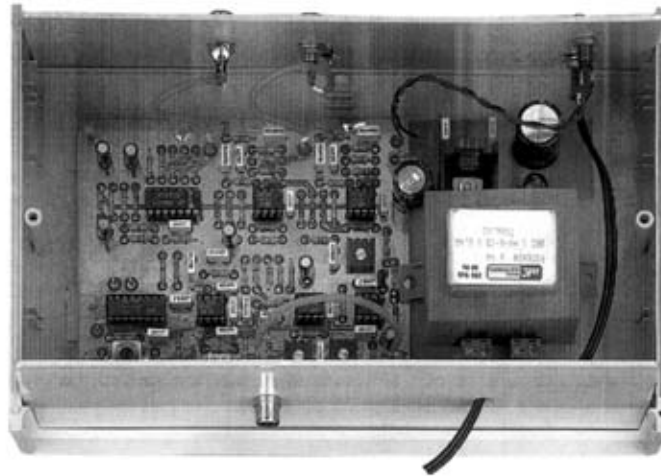
Esquema de montaje práctico de la placa LX.1248 y disposición de terminales de los semiconductores utilizados en el circuito.

MICROF.

MICROF.



Aspecto final del circuito LX.1248 y montaje en el mueble, también incluido en el kit.



Para realizar el Codificador Estéreo se necesita un **circuito impreso** de doble cara: El **LX.1248**, circuito que soporta todos los componentes. Para el montaje es importante tener presentes las siguientes consideraciones.

Zócalos: Al montar los **zócalos** para los circuitos integrados **IC1, IC2, IC3, IC4, IC5, IC6** e **IC7** hay que respetar la muesca de referencia presente en la serigrafía del circuito impreso y no utilizar mucho estaño para no provocar cortocircuitos.

Resistencias: Cuando se monten las **resistencias** que incluye el circuito (**R1-R12, R14-R18, R20-R50, R52-R58**) hay que controlar su valor óhmico, si es preciso con la ayuda de una tabla de colores. En el caso de los **trimmers horizontales (R13, R19, R51)** el valor se controla mediante la serigrafía impresa sobre su cuerpo.

Condensadores: Hay que controlar su valor por la serigrafía impresa en su cuerpo. Al montar los de **poliéster (C1, C4, C6-C10, C12-C19, C21, C25-C27, C29-C32, C34, C36-C37, C41-C42)** y los **cerámicos (C2-C3, C5, C11, C28, C33, C35)** no hay que preocuparse por la polaridad ya que carecen de ella. En cambio, al montar los condensadores **electrolíticos (C20, C22-C24, C38-C40, C43)** sí hay que tener en cuenta la polaridad de sus terminales.

Semiconductores: Al realizar el montaje del circuito integrado **IC8** ha de fijarse horizontalmente con su **aleta de refrigeración** utilizando un **tornillo** y su **tuerca**. El **punteo rectificador (RS1)** se instala con el terminal **+** orientado hacia la derecha.

Diodos LED: Al montarlos hay que respetar la polaridad, el **Ánodo (A)** es el terminal **más largo**. Este circuito incluye un **diodo LED (DL1)** que se suelda al circuito impreso a través de dos cables.

Conectores: Este circuito incluye **dos clemas de 2 polos** para la conexión de la tensión de red de 220 voltios y para el interruptor de encendido (**S1**). También se incluyen **3 conectores RCA hembra** para la conexión de las señales de entrada y para la señal de la salida codificada en estéreo que se han de instalar en los paneles del mueble a través de **cables coaxiales** (los **2 conectores de entrada** en el **panel frontal** y el conector de **salida** en el **panel trasero**). Por último, el circuito incluye **3 conectores tipo pin** utilizados como puntos de comprobación (**TP1-TP3**).

Interruptores y pulsadores: El **interruptor** de encendido (**S1**) se ha de fijar en el panel frontal del mueble, posteriormente hay conectarlo a una clema del circuito impreso (ver esquema de montaje).

Circuitos integrados con zócalo: Los integrados **IC1, IC2, IC3, IC4, IC5, IC6** e **IC7** se han de introducir en sus correspondientes zócalos haciendo coincidir las muescas de referencia en forma de **U** de los integrados con la de los zócalos.

Elementos diversos: Además de los componentes ya relacionados hay varios más: Un **transformador (T1)**, que se suelda directamente en el circuito impreso en la única posición posible, una **MF (MF1)**, al instalarla no hay que olvidar soldar las lengüetas a masa y un **cordón de alimentación** que se pasa a través de un orificio del panel trasero del mueble y se conecta a una clema (ver esquema de montaje).

MONTAJE EN EL MUEBLE: En el kit se incluye el mueble de plástico en el que se ha de fijar el circuito impreso con **4 separadores** con **base autoadhesiva**. En el panel **frontal** del mueble hay que instalar el **portaled** para **DL1**, el interruptor **S1** y los dos **bornes de entrada**. En el panel **trasero** hay que instalar el **borne de salida** y hacer pasar el **cordón de alimentación**. Los paneles del mueble **no** están perforados por lo que hay que realizar los taladros correspondientes.

AJUSTE Y PRUEBA: Para ajustar y probar el Codificador hay que disponer de un **transmisor FM** y de un **receptor estéreo FM** sintonizados en la misma frecuencia. Los pasos a seguir son los siguientes: **(1)** Girar a medio recorrido **R19** y a cuarto recorrido **R51** **(2)** Girar lentamente el núcleo de **MF1** hasta que se encienda el diodo LED estéreo del receptor **FM** **(3)** Aplicar una señal en el canal derecho y ajustar **R13-R51** de forma que solo se escuche la señal en el altavoz derecho del receptor **(4)** Aplicar una señal en el canal izquierdo y ajustar **R13-R51** de forma que solo se escuche la señal en el altavoz izquierdo del receptor.

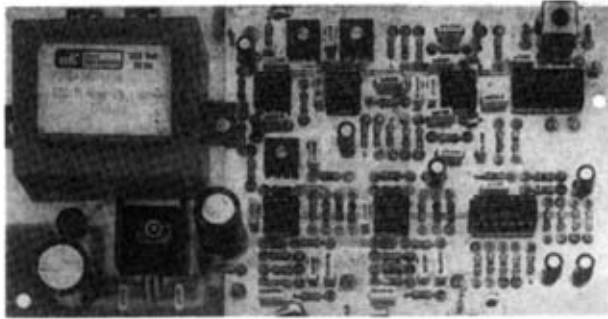
UTILIZACIÓN: La señal a aplicar a la **entrada** del **Codificador** se ha de obtener de la salida de un **preamplificador estéreo**. La **salida** del **Codificador** se aplica a la entrada un **transmisor FM** corriente con una sola toma de entrada mono, que utilizando el Codificador **transmitirá en estéreo** la señal presente en las entradas del Codificador **LX.1248**.

PRECIOS Y REFERENCIAS

LX.1248: Todos los componentes necesarios para la realización del kit, incluido el circuito impreso y el mueble contenedor	
MTK10.13	96,01 € + IVA
LX.1248: Circuito impreso	29,81 € +IVA

Estos precios no incluyen I.V.A.

Revista de aparición del kit: N.145



Aspecto final del circuito LX.1248 y montaje en el mueble, también incluido en el kit.



Para realizar el Codificador Estéreo se necesita un **circuito impreso** de doble cara: El **LX.1248**, circuito que soporta todos los componentes. Para el montaje es importante tener presentes las siguientes consideraciones.

Zócalos: Al montar los **zócalos** para los circuitos integrados **IC1, IC2, IC3, IC4, IC5, IC6** e **IC7** hay que respetar la muesca de referencia presente en la serigrafía del circuito impreso y no utilizar mucho estaño para no provocar cortocircuitos.

Resistencias: Cuando se monten las **resistencias** que incluye el circuito (**R1-R12, R14-R18, R20-R50, R52-R58**) hay que controlar su valor óhmico, si es preciso con la ayuda de una tabla de colores. En el caso de los **trimmers horizontales (R13, R19, R51)** el valor se controla mediante la serigrafía impresa sobre su cuerpo.

Condensadores: Hay que controlar su valor por la serigrafía impresa en su cuerpo. Al montar los de **poliéster (C1, C4, C6-C10, C12-C19, C21, C25-C27, C29-C32, C34, C36-C37, C41-C42)** y los **cerámicos (C2-C3, C5, C11, C28, C33, C35)** no hay que preocuparse por la polaridad ya que carecen de ella. En cambio, al montar los condensadores **electrolíticos (C20, C22-C24, C38-C40, C43)** sí hay que tener en cuenta la polaridad de sus terminales.

Semiconductores: Al realizar el montaje del circuito integrado **IC8** ha de fijarse horizontalmente con su **aleta de refrigeración** utilizando un **tornillo** y su **tuerca**. El **punteo rectificador (RS1)** se instala con el terminal **+** orientado hacia la derecha.

Diodos LED: Al montarlos hay que respetar la polaridad, el **Ánodo (A)** es el terminal **más largo**. Este circuito incluye un **diodo LED (DL1)** que se suelda al circuito impreso a través de dos cables.

Conectores: Este circuito incluye **dos clemas de 2 polos** para la conexión de la tensión de red de 220 voltios y para el interruptor de encendido (**S1**). También se incluyen **3 conectores RCA hembra** para la conexión de las señales de entrada y para la señal de la salida codificada en estéreo que se han de instalar en los paneles del mueble a través de **cables coaxiales** (los **2 conectores de entrada** en el **panel frontal** y el conector de **salida** en el **panel trasero**). Por último, el circuito incluye **3 conectores tipo pin** utilizados como puntos de comprobación (**TP1-TP3**).

Interruptores y pulsadores: El **interruptor** de encendido (**S1**) se ha de fijar en el panel frontal del mueble, posteriormente hay conectarlo a una clema del circuito impreso (ver esquema de montaje).

Circuitos integrados con zócalo: Los integrados **IC1, IC2, IC3, IC4, IC5, IC6** e **IC7** se han de introducir en sus correspondientes zócalos haciendo coincidir las muescas de referencia en forma de **U** de los integrados con la de los zócalos.

Elementos diversos: Además de los componentes ya relacionados hay varios más: Un **transformador (T1)**, que se suelda directamente en el circuito impreso en la única posición posible, una **MF (MF1)**, al instalarla no hay que olvidar soldar las lengüetas a masa y un **cordón de alimentación** que se pasa a través de un orificio del panel trasero del mueble y se conecta a una clema (ver esquema de montaje).

MONTAJE EN EL MUEBLE: En el kit se incluye el mueble de plástico en el que se ha de fijar el circuito impreso con **4 separadores** con **base autoadhesiva**. En el panel **frontal** del mueble hay que instalar el **portaled** para **DL1**, el interruptor **S1** y los dos **bornes de entrada**. En el panel **trasero** hay que instalar el **borne de salida** y hacer pasar el **cordón de alimentación**. Los paneles del mueble **no** están perforados por lo que hay que realizar los taladros correspondientes.

AJUSTE Y PRUEBA: Para ajustar y probar el Codificador hay que disponer de un **transmisor FM** y de un **receptor estéreo FM** sintonizados en la misma frecuencia. Los pasos a seguir son los siguientes: **(1)** Girar a medio recorrido **R19** y a cuarto recorrido **R51** **(2)** Girar lentamente el núcleo de **MF1** hasta que se encienda el diodo LED estéreo del receptor **FM** **(3)** Aplicar una señal en el canal derecho y ajustar **R13-R51** de forma que solo se escuche la señal en el altavoz derecho del receptor **(4)** Aplicar una señal en el canal izquierdo y ajustar **R13-R51** de forma que solo se escuche la señal en el altavoz izquierdo del receptor.

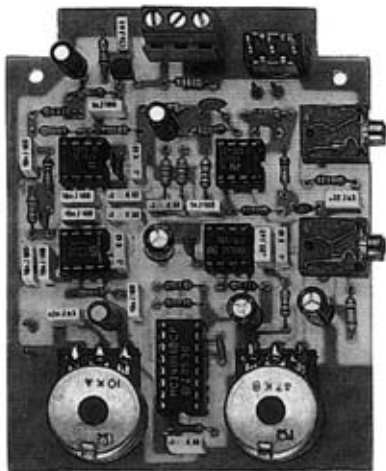
UTILIZACIÓN: La señal a aplicar a la **entrada** del **Codificador** se ha de obtener de la salida de un **preamplificador estéreo**. La **salida** del **Codificador** se aplica a la entrada un **transmisor FM** corriente con una sola toma de entrada mono, que utilizando el Codificador **transmitirá en estéreo** la señal presente en las entradas del Codificador **LX.1248**.

PRECIOS Y REFERENCIAS

LX.1248: Todos los componentes necesarios para la realización del kit, incluido el circuito impreso y el mueble contenedor
 MTK10.1396,01 € + IVA
CC.1248: Circuito impreso29,81 € +IVA

Estos precios no incluyen I.V.A.

Revista de aparición del kit: N.145



Aspecto final del circuito LX.1226 y montaje en el mueble, no incluido en el kit (hay que solicitarlo aparte).



Para realizar el Detector de Ultrasonidos se necesita un **circuito impreso** de doble cara: El **LX.1226**, circuito que soporta todos los componentes. Para el montaje es importante tener presentes las siguientes consideraciones.

Zócalos: Al montar los **zócalos** para los circuitos integrados **IC1, IC2, IC3, IC4 e IC5** hay que respetar la muesca de referencia presente en la serigrafía del circuito impreso y no utilizar mucho estaño para no provocar cortocircuitos.

Resistencias: Cuando se monten las **resistencias** que incluye el circuito (**R1-R6, R8-R22, R24-R26**) hay que controlar su valor óhmico, si es preciso con la ayuda de una tabla de colores. En el caso de los **potenciómetros** (**R7, R23**), que se han de soldar directamente al circuito impreso como se muestra en el esquema de montaje práctico, el valor se controla mediante la serigrafía impresa sobre su cuerpo.

Condensadores: Hay que controlar su valor por la serigrafía impresa en su cuerpo. Al montar los de **poliéster** (**C2-C3, C5-C7, C11-C15, C17-C20, C23, C25-C27, C32**) y los **cerámicos** (**C8-C10, C21-C22, C24, C30**) no hay que preocuparse por la polaridad ya que carecen de ella. En cambio, al montar los condensadores **electrolíticos** (**C1, C4, C16, C28-C29, C31**) sí hay que tener en cuenta la polaridad de sus terminales.

Semiconductores: Al realizar el montaje del **FET (FT1)** hay que soldarlo respetando la disposición de terminales, para lo cual hay que orientar su lado plano tal y como se indica en el esquema de montaje práctico.

Conectores: Este circuito incluye una **clema de 3 polos** para la conexión del **micrófono ultrasónico** y **2 conectores hembra** tipo **Jack** utilizados para la conexión de **auriculares** y de una **grabadora**. También

incluye un **portapilas de 9 voltios** cuyos cables de conexión se sueldan directamente al impreso, teniendo cuidado en respetar su polaridad (cable rojo al positivo y cable negro al negativo).

Interruptores y pulsadores: El **interruptor** de encendido (**S1**) se ha de fijar directamente en el circuito impreso, tal y como se muestra en el esquema de montaje práctico.

Circuitos integrados con zócalo: Los integrados **IC1, IC2, IC3, IC4 e IC5** se han de introducir en sus correspondientes zócalos haciendo coincidir las muescas de referencia en forma de **U** de los integrados con la de los zócalos.

Elementos diversos: El kit incluye un **micrófono ultrasónico (MIC)** que se instala conectándolo a la clema de tres polos, por tanto sin realizar ninguna soldadura.

MONTAJE EN EL MUEBLE: En el kit **no** incluye mueble contenedor, si se desea se puede pedir el contenedor de plástico **MO.1226** que incluye un **panel serigrafiado**. Antes de fijar el circuito impreso dentro del mueble hay que realizar en el mueble los **orificios** para los potenciómetros, el interruptor y los conectores.

AJUSTE Y PRUEBA: Este circuito no precisa ningún ajuste.

UTILIZACIÓN: Detectar ultrasonidos es muy sencillo con este Detector, de hecho basta con dirigir la cápsula microfónica hacia **cualquier dirección** y luego girar el potenciómetro de ajuste de la **Frecuencia (R7)** hasta percibir algún sonido. Como prueba se puede hacer mover una pulsera metálica, aunque aparentemente no se produce ningún ruido, con los auriculares se escuchará un ruido ensordecedor.

PRECIOS Y REFERENCIAS

LX.1226: Todos los componentes necesarios para la realización del Detector de Ultrasonidos, únicamente excluido el mueble contenedor	60,10 € + IVA
MO.1226: Mueble de plástico con panel serigrafiado.....	19,65 € + IVA
CC.1226: Circuito impreso	15,63 € +IVA

Revista de aparición del kit: N.141

RADIO RHIN

**EL
MAYOR**

AUTOSERVICIO

de componentes electrónicos

- TV, VIDEO Y SONIDO PROFESIONAL.
- ANTENAS, SEMICONDUCTORES, KITS, SONORIZACIÓN...ETC.
- CABLES Y CONEXIONES INFORMÁTICAS.

RADIO RHIN



ALAMEDA URQUIJO 32
48010 BILBAO

94 443 17 04

Fax: 94 443 15 50

e-mail: radorhin@elec.euskalnet.net