



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

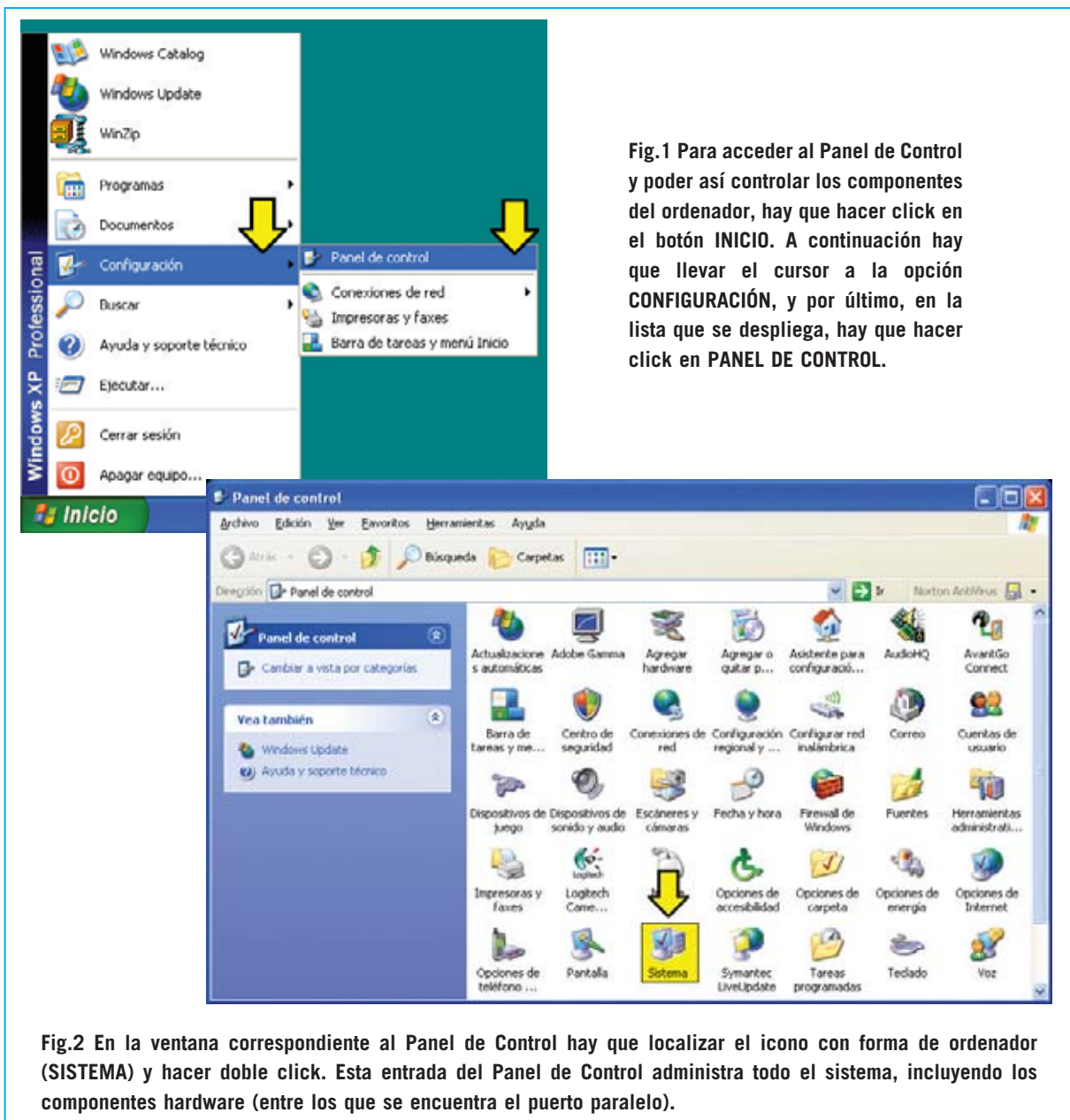


Fig.1 Para acceder al Panel de Control y poder así controlar los componentes del ordenador, hay que hacer click en el botón INICIO. A continuación hay que llevar el cursor a la opción CONFIGURACIÓN, y por último, en la lista que se despliega, hay que hacer click en PANEL DE CONTROL.

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).

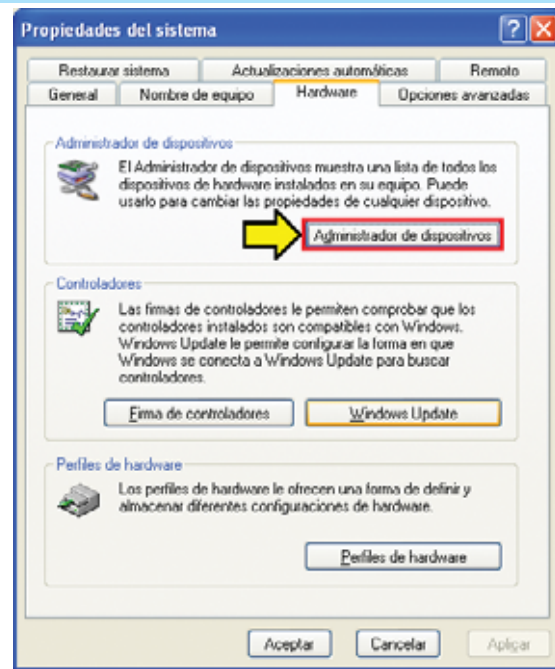


Fig.3 La ventana SISTEMA dispone de varias entradas diferentes a las que se accede a través de las pestañas superiores. Para acceder a la administración de los dispositivos hay que seleccionar la pestaña HARDWARE y, a continuación, el botón ADMINISTRADOR DE DISPOSITIVOS.

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).

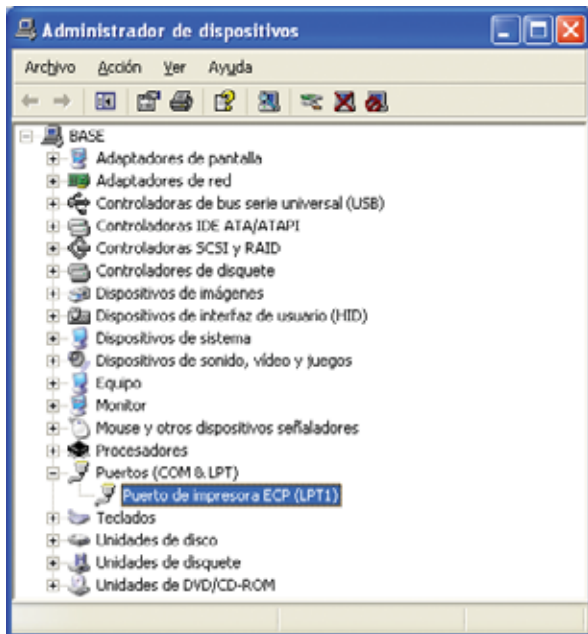


Fig.5 Aspecto del menú INTEGRATED PERIPHERALS en el Setup de las BIOS AMI y AWARD. La opción correspondiente al ajuste del modo del puerto (PARALLEL PORT MODE) está resaltada.

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

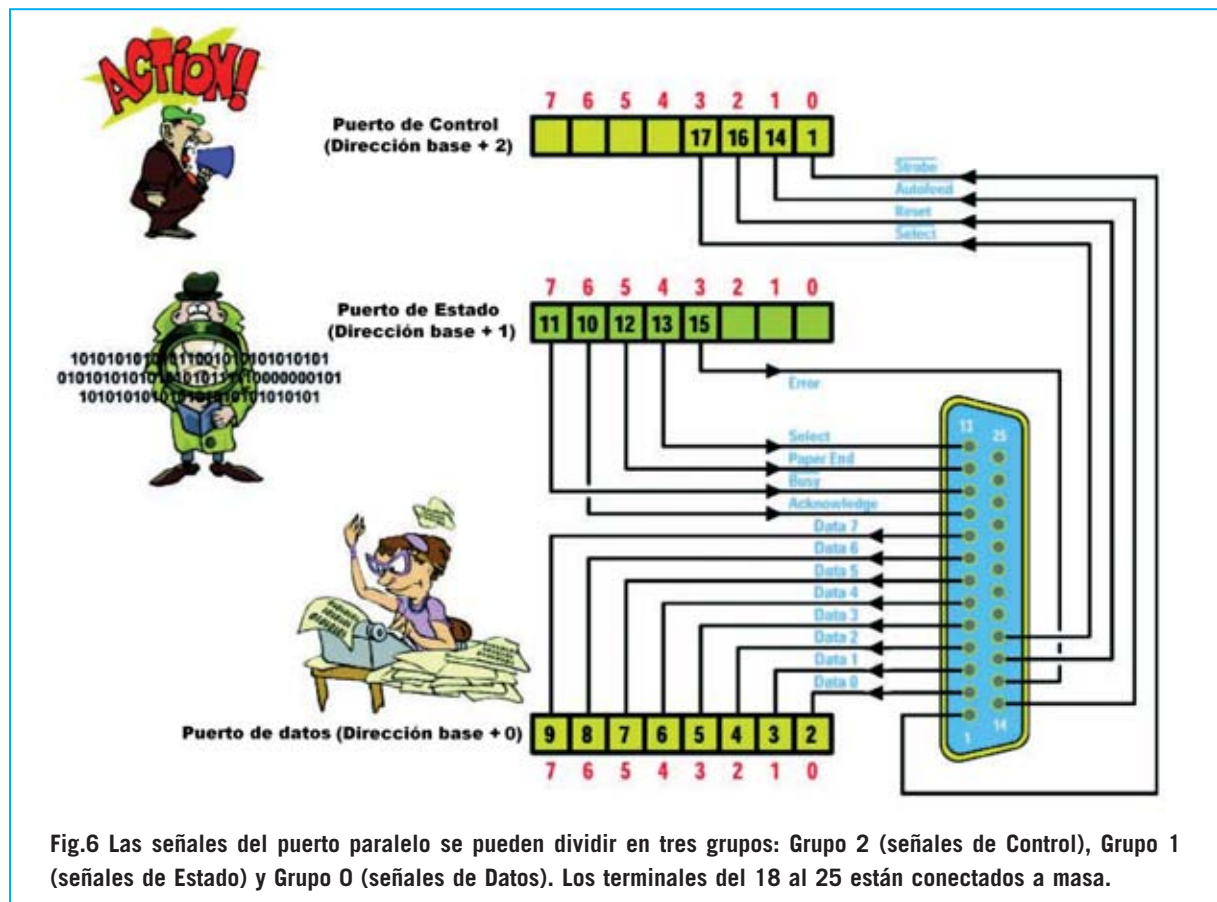


Fig.6 Las señales del puerto paralelo se pueden dividir en tres grupos: Grupo 2 (señales de Control), Grupo 1 (señales de Estado) y Grupo 0 (señales de Datos). Los terminales del 18 al 25 están conectados a masa.

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**.

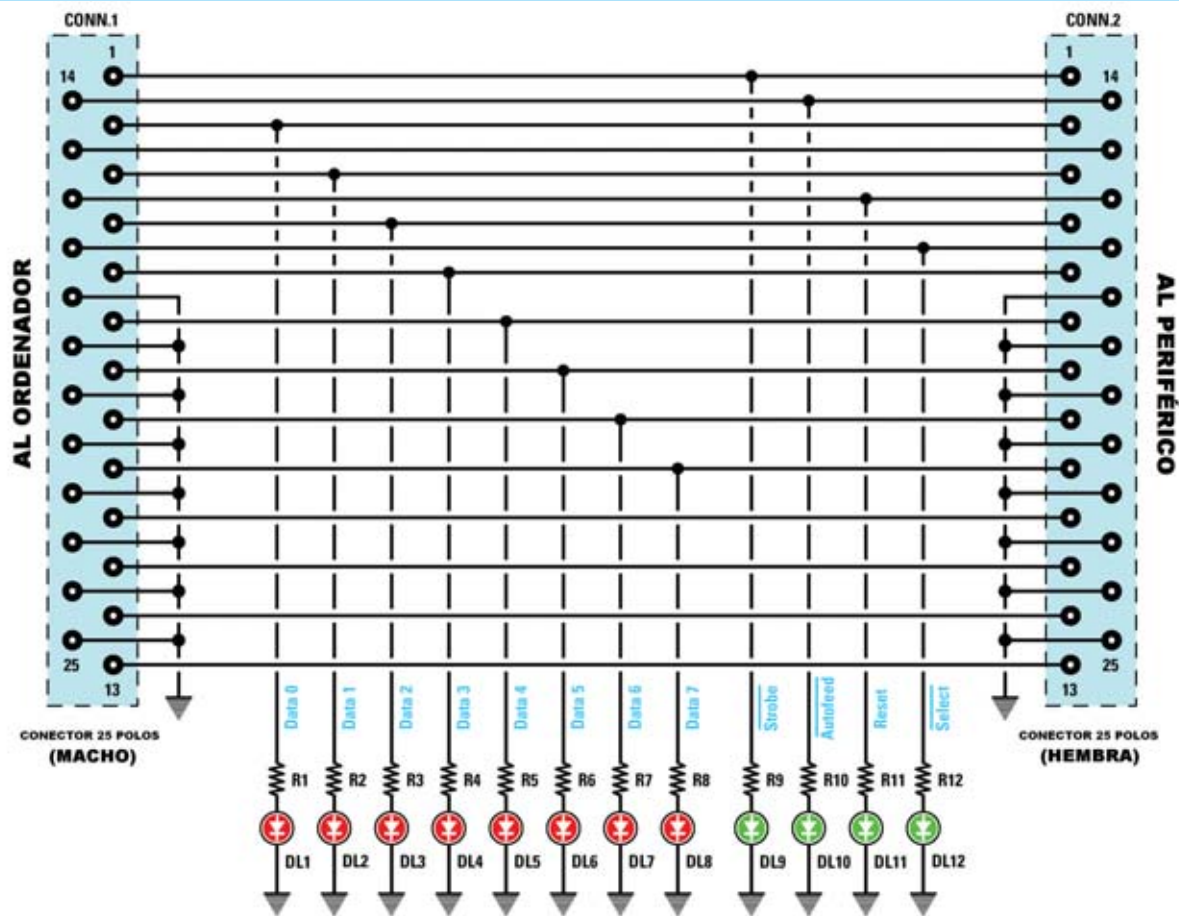


Fig.7 Esquema eléctrico del Téster LX.1588, útil para controlar en tiempo real el funcionamiento del puerto paralelo. Los 8 diodos LED rojos (DL1-DL8) están conectados al Grupo 0 para monitorizar las señales de datos, mientras que los 4 diodos LED verdes monitorizan las señales del Grupo 2 (Control).

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

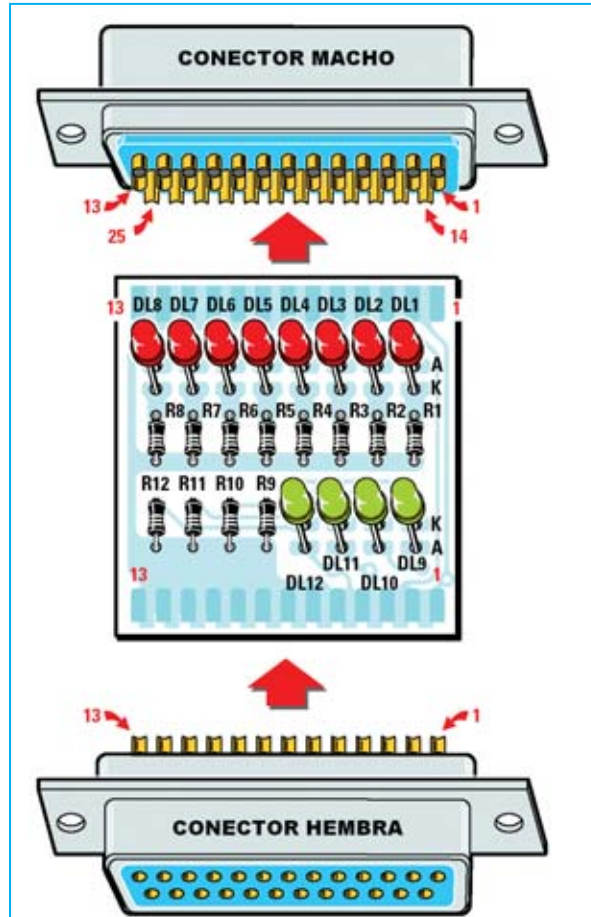


Fig.9 Esquema práctico de montaje del Téster para puerto paralelo LX.1588. Los diodos LED rojos se montan orientando el Ánodo hacia la parte superior mientras que los diodos LED verdes se montan orientando el Ánodo hacia la parte inferior.

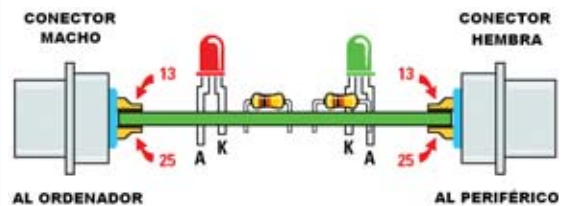


Fig.10 Como se puede observar el circuito impreso se aloja entre las dos filas de terminales de los conectores. Los terminales de los conectores se sueldan directamente a las pistas del circuito impreso.

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.

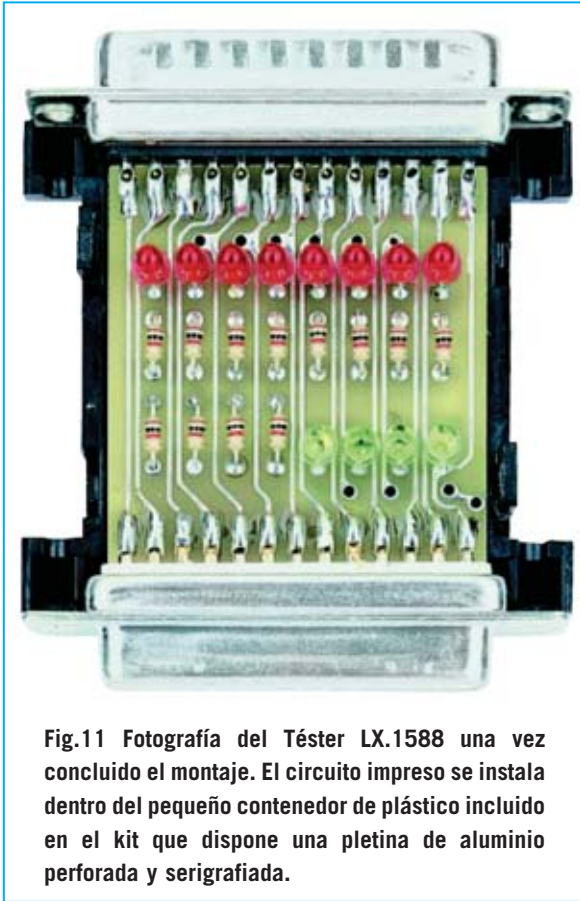


Fig.11 Fotografía del Téster LX.1588 una vez concluido el montaje. El circuito impreso se instala dentro del pequeño contenedor de plástico incluido en el kit que dispone una pletina de aluminio perforada y serigrafiada.

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.

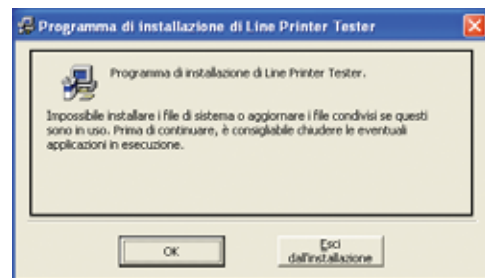


Fig.16 Para comenzar la instalación del programa Line Printer Tester hay que hacer click en el botón OK.

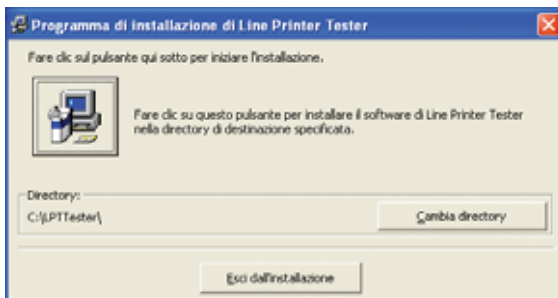


Fig.17 El directorio predeterminado para la instalación es C:\LPTTester\. Para continuar hay que hacer click en el icono con forma de ordenador.



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.

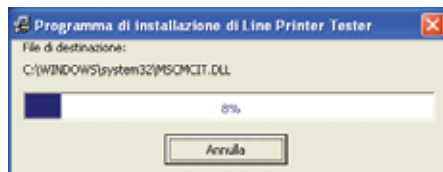


Fig.19 En la fase de copia de archivos se muestra una barra de progreso de color azul.

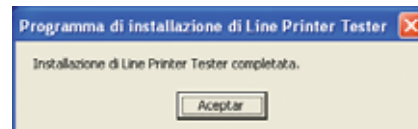


Fig.20 LA instalación del programa LPT Tester ha concluido. Para terminar hay que hacer click en ACEPTAR.

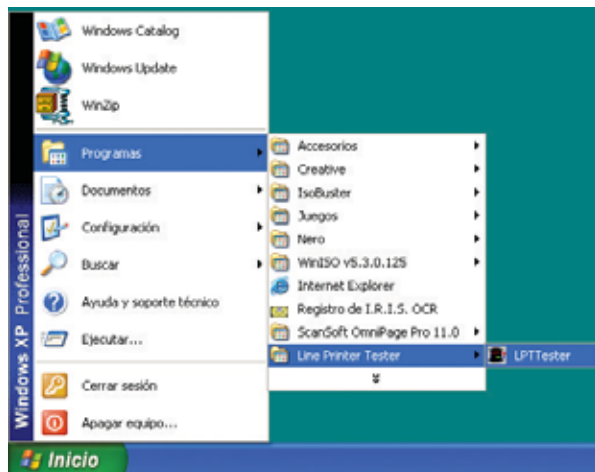


Fig.21 Para ejecutar el programa LPT Tester hay que hacer click en el botón INICIO. A continuación hay que llevar el cursor sobre PROGRAMAS y, en la lista desplegada, hay que localizar la entrada LINE PRINTER TESTER. Por último hay que hacer click sobre LPT TESTER.

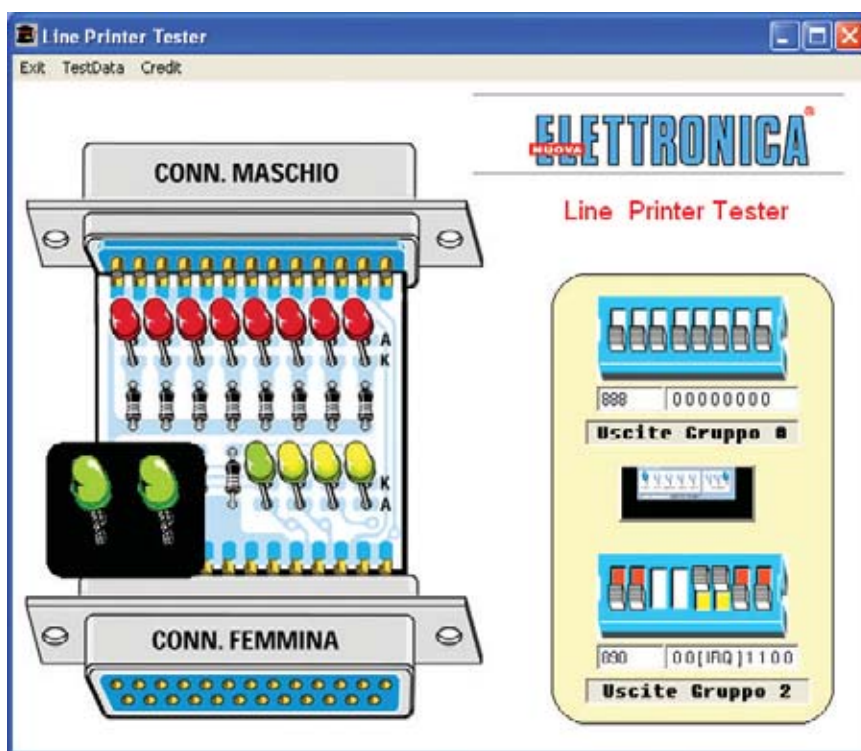
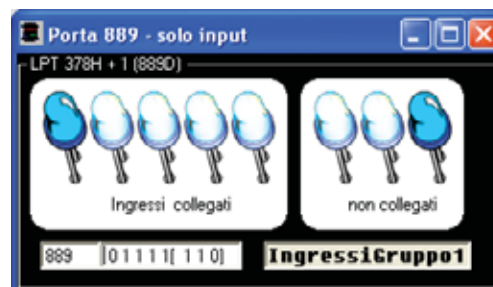


Fig.22 Así se presenta la ventana principal del programa LPT Tester. En la parte izquierda está representado el circuito impreso con sus diodos LED, mientras que en la parte derecha se encuentran unos dip-switch que controlan las señales del Grupo 0 y del Grupo 2. Actuando sobre estos dip-switch se controlan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

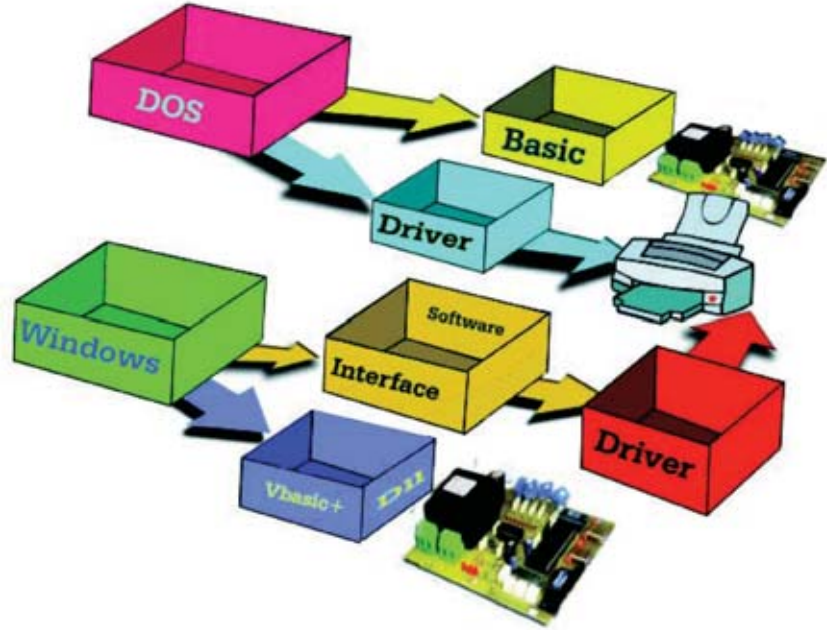
Si tenéis **Basica** de IBM: **BASICA** y pulsar **ENTER**.

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

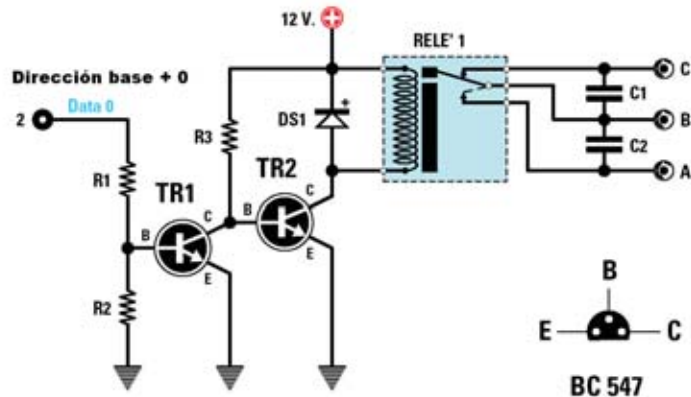
REM en el variable **Buffer** encontraremos

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

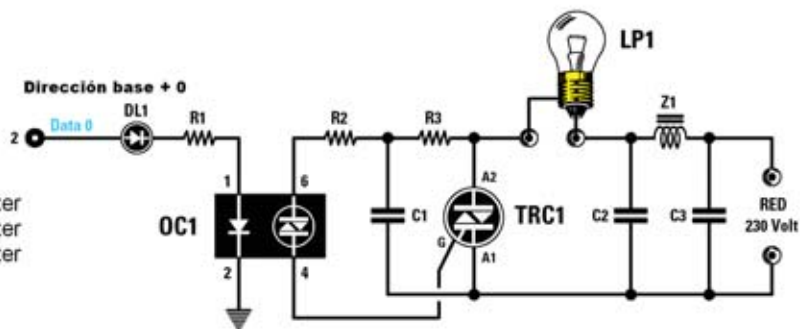
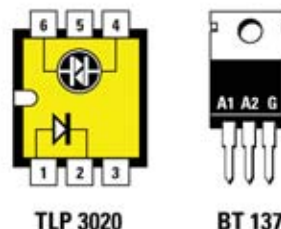


Fig.26 Esquema eléctrico de un circuito que controla cargas a 230 voltios de tipo resistivo desde el puerto paralelo del ordenador. Este esquema utiliza un optoacoplador TLP.3020, cuyas conexiones se muestran junto a las del TRIAC BT.137.



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

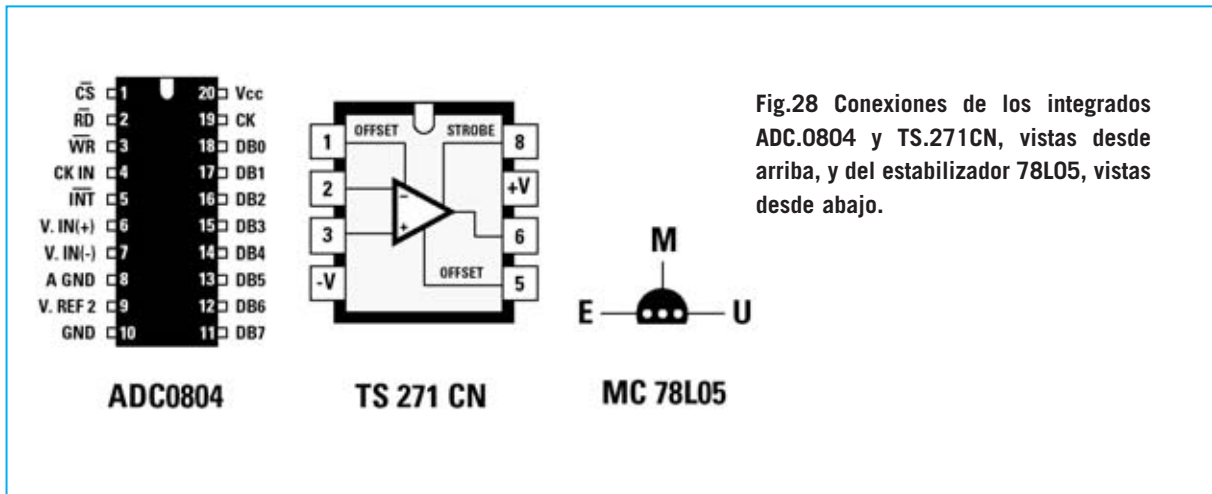
Con el sistema operativo Windows **no** basta con tener un programa como **Visual Basic** y escribir las mismas instrucciones que hemos usado con el intérprete BASIC para DOS, ya que el puerto y el periférico de impresión **no** dialogan directamente con la BIOS del ordenador sino a través de un **programa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

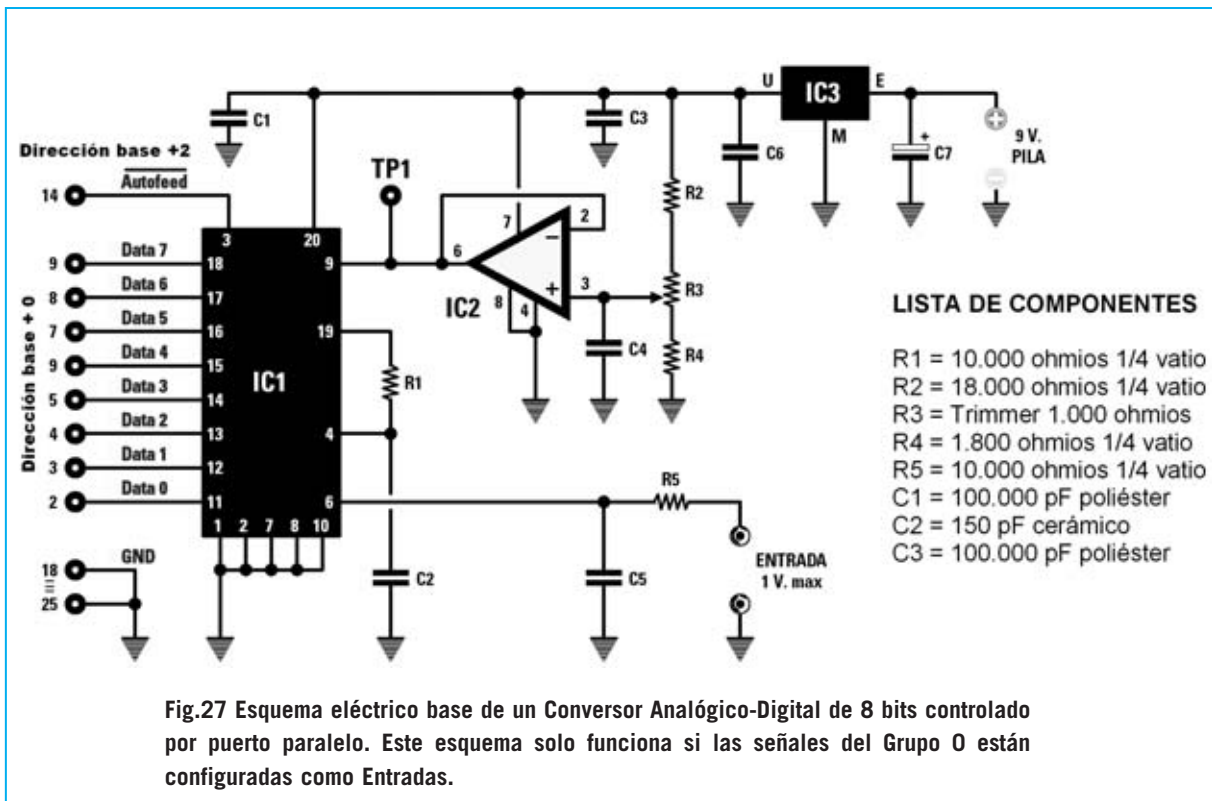


Fig.29. Para ejecutar el programa de conversión A/D hay que hacer click en el botón INICIO. A continuación hay que llevar el cursor sobre PROGRAMAS y localizar la entrada TEST ADC0804. Por último hay que hacer click sobre TEST ADC0804.



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).

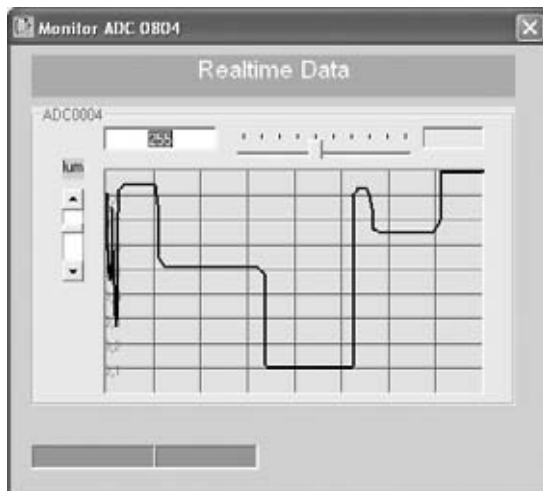


Fig.31 En la ventana REALTIME DATA los valores de tensión que llegan del Conversor A/D se muestran como en un osciloscopio.

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 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 Convertor 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.