

PUERTO PARALELO

Comúnmente denominado puerto de impresora original o puerto centronix este conector permite entrar y salir de una computadora PC señales las cuales podemos emplear para muchas cosas más que simplemente imprimir sobre papel. Toda computadora compatible con PC dispone de al menos un puerto paralelo el cual es compatible con la norma original estándar o SPP. Adicionalmente, algunos mas nuevos permiten el uso de datos bidireccionales e incluso de características especiales definidas por las normas EPP y ECP, de los cuales haremos una breve reseña más adelante en esta misma nota.

Retomando el SPP debemos indicar que, con placas de expansión se pueden disponer de hasta 3 puertos que el sistema operativo denomina LPT. Se los numera desde 1 hasta 3 quedando LPT1, LPT2 y LPT3. Como todo dispositivo, el puerto paralelo dispone de una dirección de memoria base que puede ser 378, 278 o 3BC. Es por esta dirección por la que el usuario (a través del programa) se comunica y emplea el puerto paralelo.

Físicamente, el puerto paralelo se conecta por medio de un conector DB25 hembra que habitualmente se ubica en el panel posterior del equipo.

Técnicamente el puerto paralelo dispone de 8 líneas digitales de salida, las cuales llamaremos D0 a D7, cuyos niveles posibles son 0v (para bajo) y 5v (para alto), cumpliendo con el estándar de la familia lógica TTL. Estas líneas están conectadas a los pines 2 al 9 para los bits 0 al 7 respectivamente. Estas ocho líneas conforman el comúnmente llamado bus de datos del puerto. Pero aquí no termina todo. Hay dos buses más que permiten ingresar y salir desde y hacia la computadora llamados buses de estado y de control.

El bus de estado se empleaba originalmente para censar justamente el estado en el que se encontraba la impresora para saber, entre otras cosas, si está encendida, si está en línea, si dispone de espacio para recibir datos, si hay papel, etc. Pero puede ser empleado para ingresar datos hacia la computadora desde el mundo exterior haciendo, por ejemplo, que el programa decida su rumbo y por ende las acciones a tomar en función a los datos que por aquí ingresen. Este puerto, a diferencia del de datos, dispone de sólo 5 líneas digitales de entrada partiendo desde el bit 3 y hasta el bit 7. Por lo general a estas líneas se las denomina S3 a S7 y la asignación de pines se la puede ver en la tabla de más abajo. La dirección de memoria asignada para comunicarse con este bus es igual a la dirección base (la del bus de datos) más 1. O sea, si el puerto base es 378, este puerto es 379. Si el base es 278, este es 279. Y, si el puerto base es 3BC, este será 3BD. Un dato a resaltar es que el bit 7 de este puerto (conectado al terminal 11 del conector) presenta un estado lógico inverso. Esto significa que, si en el pin 11 se colocan 5v (nivel lógico alto) el bit al ser leído presentará un cero. Y, por consiguiente, si el pin 11 es puesto a masa (nivel lógico bajo) al leer el bit se verá un uno.

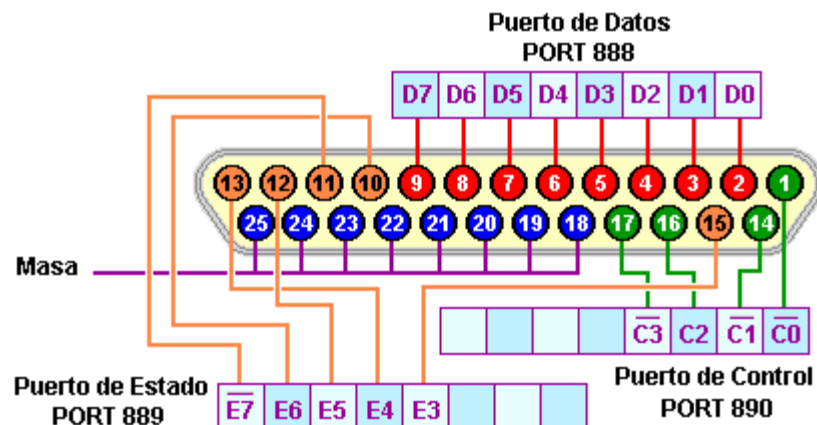
En tanto, el bus de control, al igual que el de datos, dispone de líneas de salida desde la computadora hacia el exterior. Las líneas también son digitales pero en este caso son solo cuatro, las de menor pero que van del bit 0 al 3. Estas cuatro líneas son generalmente llamadas como C0 a C4 y se pueden controlar por medio de la dirección de memoria base (la del bus de datos) más 2. De esta forma si la dirección base es 378, este bus es 37A. Si la dirección base es 278, la dirección de este bus será 27A. Y, si la dirección base es 3BC, la

dirección de este bus será 3BE. En este bus hay tres bits cuyos niveles lógicos se encuentran invertidos (bit0, bit1 y bit 3) y sólo uno que presenta un un estado lógico normal.

Resumiendo podemos decir que un puerto paralelo estándar o SPP permite salir desde la computadora hacia el exterior por medio de 12 líneas o bits y posibilita entrar a la misma por medio de 5 líneas o bits.

En todos los casos cada terminal puede proporcionar un máximo de 1mA de corriente. Superado este miliamperio no se garantiza el nivel lógico TTL de la señal presente e incluso de sobrecargarlo demasiado de puede dañar físicamente el puerto. Una forma de utilizar en forma segura el puerto es conectando a cada terminal de salida un resistor de 100 ohms en serie y luego un buffer octal como el integrado 74LS245. Esto permite, por un lado, aislar el puerto del dispositivo y, por el otro, efectuar una adecuada carga de corriente al sistema.

BUS	PIN	TTL	E/S
DATOS	2	NORmal	Salida
	3	NORmal	Salida
	4	NORmal	Salida
	5	NORmal	Salida
	6	NORmal	Salida
	7	NORmal	Salida
	8	NORmal	Salida
	9	NORmal	Salida
ESTADO	10	NORmal	Entrada
	11	INVertido	Entrada
	12	NORmal	Entrada
	13	NORmal	Entrada
	15	NORmal	Entrada
CONTROL	1	INVertido	Salida
	14	INVertido	Salida
	16	NORmal	Salida
	17	INVertido	Salida



En la tabla se escribe INV para indicar que el nivel lógico invertido (o sea que un bit 1 representa una tensión de 0v y un bit 0 representa una tensión de 5v). NOR indica un nivel lógico normal, quedando un bit bajo en 0v y uno alto en 5v. E significa entrada (por este pin entra tensión y por ende datos hacia la computadora). S significa salida. --- indica que el presente valor no está establecido o implementado.

Hasta aquí hablamos del aspecto electrónico y mecánico del puerto. Ahora veremos el aspecto relativo al software y la forma de escribir adecuadamente el código de los algoritmos que controlaran las líneas del puerto.

El bus de datos del puerto se comporta como un latch consultable. Esto quiere decir que enviando un byte a la dirección de memoria los pines del conector quedarán a 5v y 0v en función a los bits de dicho byte. Y esto quedará así hasta que un nuevo dato sea enviado. Si en lugar de escribir leemos la dirección obtendremos un byte cuyos bits serán ceros y unos en función del estado lógico de las líneas. Ese dato obtenido será el último inmediato enviado y no será alterado bajando o subiendo las líneas de datos. En otras palabras, si se escribe FF en el bus de datos del puerto y se colocan LED's entre cada pin y masa todos estarán encendidos. Si leemos el bus de datos obtendremos un byte cuyo valor será FF. Pero si manualmente ponemos a masa las líneas del bus de datos en el conector y nuevamente leemos, el valor presente seguirá siendo FF. Es más, esta práctica en equipos antiguos puede dañar la electrónica de salida del puerto.

Siguiendo con las salidas, el bus de control sólo posee cuatro líneas (las correspondientes a los bits más bajos, las cuales se manejan de la misma forma que el bus de datos. Hay dos salvedades que aclarar. La primera es que no conviene poner en 1 los cuatro bits más altos de este bus (los que físicamente no están implementados). Esto se resuelve impidiendo que el soft envíe cualquier dato superior a 15 al bus de control del puerto. Y la segunda es que, como tres de las cuatro líneas presentan niveles lógicos invertidos, para que todos los pines del conector estén a 0v el dato presente en este puerto deberá ser 0B en hexadecimal (11 en decimal). Esto se deduce de lo siguiente: Los pines asignados a los bits de control 0, 1 y 3 presentan 5v cuando en el bus los mismos bits están en 0 y presentan 0v cuando están en 1. Por ende, para que en las salidas los pines no presenten tensión deberán ponerse esos bits en 1. Este bus, al igual que el de datos, es un latch consultable, corriendo con las mismas normas explicadas arriba.

Concluyendo, el bus de estado puede ser leído, y por consiguiente saberse el estado de las líneas en sus pines, con solo hacer una consulta o lectura a la dirección de memoria asignada. De esta forma, inmediatamente hecha la consulta, se censa el estado de las líneas, se invierte el nivel lógico de la correspondiente al bit 7 (pin 11) y se coloca el dato a disposición del usuario. Este bus no es latcheado por lo que, para un correcto funcionamiento, el dato a leer debe ser sostenido tanto como demore la instrucción de consulta al puerto. Escribir en este bus no produce efecto alguno. Cabe aclarar que las tensiones presentes en los pines deben respetar el estándar de la familia TTL, o sea, 0v (masa) y 5v (tensión). Dado que los tres primeros bits (los menos significativos) no están implementados es conveniente no interpretarlos. Para ello lo mejor es primero leer el dato y seguidamente hacerle un AND con el decimal 248 (00011111 en binario o 1F en hexa). De esta forma el dato resultante estará formado exclusivamente por los bits 3, 4, 5, 6 y 7.

Como ya adelantamos el sistema operativo asigna a los puertos paralelos la sigla LPT, que parte de la contracción de **Line PrinTer**. El primer dispositivo recibe el número 1 (LPT1) y el último el 3 (LPT3). Pero no siempre LPT1 es 378. En los ya obsoletos adaptadores de vídeo MGP ó Hércules que incluían un puerto paralelo con dirección base 3BC el puerto era visto por el DOS como LPT1, siguiéndole el 378 con el indicador LPT2 y por último el LPT3 en la dirección base 378. En los equipos con puerto paralelo on-board el puerto LPT1 generalmente es 378 y se puede reasignar a 278 ó 3BC utilizando la utilidad de configuración (CMOS Set-Up).

Hay que tener en cuenta lo siguiente al momento de desarrollar alguna aplicación con este puerto: Cuando se enciende la computadora y se arranca el sistema el puerto paralelo no permanece con todos sus terminales en un estado determinado sino que, aleatoriamente, los bits de los tres buses adoptan niveles azarosos. En algunas computadoras (las pocas) durante el POST (período de arranque del hardware donde se comprueba el correcto funcionamiento general del sistema) el puerto paralelo es utilizado para sacar datos relativos a los pasos de las pruebas del hardware permitiendo corregir problemas incluso sin tener vídeo.

Lo anterior, si bien no parece demasiado importante, es crucial en sistemas donde el dato enviado puede generar conflictos o problemas. Un caso sería el ejemplo de abajo:

Si al bit 0 del bus de datos conectamos un relé (con toda la electrónica necesaria para manejarlo) que accione un motor haciéndolo girar en un sentido y en el bit 1 colocamos otro relé (con todos sus componentes) que haga al mismo motor girar hacia el otro lado tendríamos la posibilidad de manejar por soft hacia qué lado queremos hacer girar el motor o si queremos dejarlo quieto. Esto podría hacerse muy fácil con dos relés. El primero asigna tensión al motor en una polaridad y el segundo en la polaridad inversa al primero. Pero si encendemos los dos relés al mismo tiempo quedarían aplicadas al motor polaridad en ambos sentidos, cortocircuitando las vías de tensión y produciendo cualquier clase de desastre.

Por lo comentado arriba deberán tomarse las medidas necesarias tanto por soft como por hard para que esta clase de problemas no ocurran.

Otro posible conflicto puede ocurrir si una aplicación intenta imprimir sobre el puerto.

Con el tiempo y la lógica evolución de la tecnología surgió el estándar EPP, que significa **Enhanced Parallel Port** o Puerto Paralelo Mejorado. Este utiliza la misma disposición de buses, el mismo conector y la misma asignación de pines que el SPP sólo que en este caso el bus de datos puede tanto enviar información hacia fuera de la computadora como recibirla desde allí.

Por último, y con el surgimiento de la necesidad de alta velocidad y capacidad de transferencia de datos por este puerto, surgió el ECP, que significa **Extended Computer Port** ó Puerto Extendido de Computadora. Este es igual al EPP pero adicionalmente el bus de datos es bidireccional permanente y, además, posee acceso directo a memoria e interrupción programable del procesador.