

# Periféricos Interfaces y Buses

- I. Arquitectura de E/S
- II. Programación de E/S
- III. Interfaces de E/S de datos
- IV. Dispositivos de E/S de datos**  
 Dispositivos de entrada de datos (teclado, escáner, tablas digitalizadoras, cámaras). Dispositivos de presentación de datos (monitores, tarjetas gráficas, impresoras). Otros dispositivos de E/S (sensores, actuadores, controladores de líneas analógicas y digitales).
- V. Buses
- VI. Controladores e interfaces de dispositivos de almacenamiento
- VII. Sistemas de almacenamiento

## Dispositivos de E/S de datos

### Bloque II (actividad desarrollada por los alumnos)

- **ratón**
- impresoras
- monitor táctil

# El Ratón

## Interfaz PS/2 – Modelo de Programación

Alberto Pérez Pérez

### Historia

El **ratón** o **mouse** es un dispositivo apuntador, generalmente fabricado en plástico que detecta su movimiento relativo en dos dimensiones por la superficie plana en la que se apoya, reflejándose habitualmente a través de un puntero o flecha en el monitor. Hoy en día es un elemento imprescindible en un equipo, pese a la aparición de otras tecnologías, y la práctica ha demostrado que tendrá todavía muchos años de vida útil.

Fué diseñado por Douglas Engelbart y Bill English durante los años 60 en la Universidad de Stanford. Más tarde fue mejorado en los laboratorios de Palo Alto de la compañía Xerox. Surgió dentro de un proyecto que buscaba aumentar el intelecto humano mejorando la comunicación entre el hombre y la máquina. Con su aparición, logró también dar el paso definitivo a la aparición de los primeros interfaces gráficas de usuario.

La primera maqueta se construyó de manera artesanal, de madera, y se patentó con el nombre de "X-Y Position Indicator for a Display System". El nombre de ratón se lo dio el equipo de la Universidad de Stanford durante su desarrollo, ya que su forma y su *cola* recuerdan a un roedor.

A pesar de su aspecto arcaico, su funcionamiento básico sigue siendo igual hoy en día. Encajaba bien en la mano y disponía de dos ruedas metálicas que, al desplazarse por la superficie, movían dos ejes: uno para controlar el movimiento vertical del cursor en pantalla y el otro para el sentido horizontal, contando además con un botón rojo en su parte superior.



Por primera vez se lograba un intermediario directo entre una persona y la computadora. A diferencia del teclado, cualquiera podía aprender a manejar sin apenas conocimientos previos.

En 1981 aparece la primera computadora con ratón incluido, la *Xerox Star 8010*. Este era fundamental para la nueva y potente interfaz gráfica que dependía de este periférico, y que fue, a su vez, otra revolución. Posteriormente, surgieron otras computadoras que también incluyeron el periférico (Commodore Amiga, el Atari ST, Apple Lisa). Dos años después, Microsoft, dio a conocer su propio diseño, disponible además con las primeras versiones del procesador de texto Word. Su precio elevado y la falta de un sistema operativo que realmente lo aprovechara, hizo que pasara completamente desapercibido.

Tras la aparición del Macintosh en 1984 este periférico se popularizó. Su diseño y creación corrió a cargo de nuevo de la Universidad de Stanford, cuando Apple en 1980 pidió a un grupo de jóvenes un periférico seguro, barato y que se pudiera producir en serie.

Si bien existen muchas variaciones posteriores, algunas innovaciones recientes y con éxito han sido el uso de una *rueda* central o lateral, el sensor de movimiento óptico por diodo LED, ambas introducidas por Microsoft en 1996 y 1999 respectivamente, o el sensor basado en un láser no visible del fabricante Logitech, que es actualmente una de las mayores empresas dedicadas a la fabricación y desarrollo de estos periféricos.



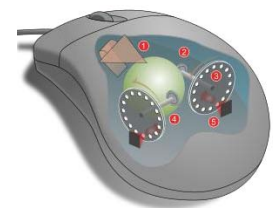
## Funcionamiento

Su funcionamiento principal depende de la tecnología que utilice para capturar el movimiento

- **Por mecanismo**

### Mecánicos

Tienen una gran bola de plástico en su parte inferior para mover dos ruedas que generan pulsos en respuesta al movimiento de éste sobre la superficie.



### Ópticos

Carece de la bola de goma evitando así el frecuente problema de la acumulación de suciedad. Es uno de los más modernos y prácticos actualmente. Puede ofrecer un límite de 800 ppp. Se basa en un sensor óptico que fotografía la superficie detectando las variaciones entre sucesivas fotografías.

### De láser

Más sensible y preciso, aconsejable para diseñadores gráficos y los jugadores de videojuegos. Detecta el movimiento mediante un láser con resoluciones a partir de 2000 ppp, con aumento significativo de la precisión y sensibilidad.



## Trackball

El concepto de *TrackBall* es una idea novedosa que parte del hecho de que se debe mover el puntero y no el dispositivo. Presentar una bola, de tal forma que cuando se coloque la mano encima se pueda mover mediante el dedo pulgar, sin necesidad de desplazar toda la mano. De esta manera se reduce el esfuerzo y la necesidad de espacio. A algunas personas, sin embargo, no les termina de resultar realmente cómodo. Este tipo ha sido muy útil por ejemplo en la informatización de la navegación marítima.



- **Por conexión**

### Por cable

Es el formato más popular y más económico, aunque existen multitud de características añadidas que pueden elevar su precio. Por ejemplo si hacen uso de tecnología láser como sensor de movimiento. Actualmente se distribuyen con dos tipos de conectores posibles, tipo USB y PS/2; antiguamente también era popular usar el puerto serie.

### Inalámbrico

En este caso el dispositivo carece de un cable que lo comunique con la computadora, en su lugar utiliza algún tipo de tecnología inalámbrica. Para ello requiere un receptor de la señal inalámbrica que produce, mediante baterías, el ratón. El receptor normalmente se conecta a la computadora por USB, o por PS/2. Según la tecnología inalámbrica usada pueden distinguirse varias posibilidades: *Radio Frecuencia*, que es el tipo más común y económico de este tipo de tecnologías, *Infrarrojo*, popular también entre los controles o mandos remotos o *Bluetooth* que es la tecnología más reciente.

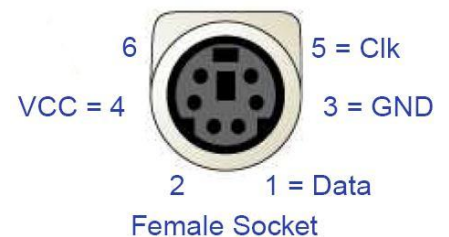
5



## Interfaz PS/2

Hay muchos tipos de dispositivos señaladores disponibles para PC, como los touchpad, pizarras electrónicas, etc. Prácticamente todos estos dispositivos se comunican con el computador utilizando las interfaces USB o PS/2.

La interfaz de ratón *PS/2* usa un protocolo bidireccional en serie para transmitir información sobre el movimiento y la pulsación de los botones hasta el controlador auxiliar de dispositivo del PC. Este envía a su vez una serie de comandos al ratón para establecer la tasa de respuesta, la resolución, etc. También proporciona al ratón energía (5v).



El conector de ratón es un conector de 6 pines.

## La interfaz eléctrica

Vcc/Ground proporcionan energía al ratón (entre 4,5v y 5,5v). El ratón no debe recibir más de 274 mA desde el PC y debe tenerse cuidado con los glitches. Estos pueden ser provocados por una conexión en caliente del ratón.

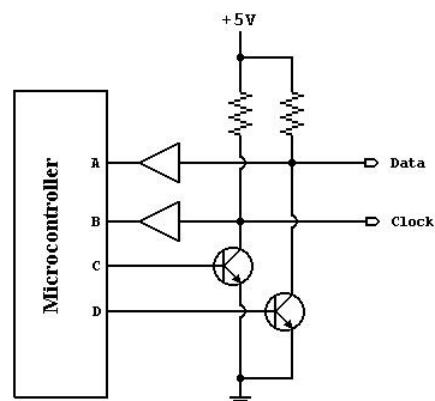
6



## Comunicación: Descripción General

El PS/2 implementa un protocolo bidireccional en serie y síncrono. El bus está en estado inactivo cuando ambas líneas están en nivel alto (colector abierto). Este es el único estado en que el ratón puede transmitir datos. El PC tiene el control del bus y puede inhibir la comunicación en cualquier momento poniendo la línea de reloj baja.

El dispositivo siempre genera la señal de reloj. Si el PC quiere enviar algún dato, debe primero inhibir la comunicación desde el dispositivo bajando la señal de reloj. Después baja la señal data y libera el reloj. Este es el estado "Request-to-Send" e indica al dispositivo para que inicie la generación de pulsos de reloj.



### Estados del bus

Data = alto, Clock = alto: *Estado inactivo.*

Data = alto, Clock = bajo: *Comunicación Inhibida.*

Data = bajo, Clock = alto: *Host Request-to-Send*



Todos los datos son **transmitidos** byte a byte y cada uno de ellos es enviado en un **paquete** de 11 o 12 bits:

- 1 bit de inicio. Siempre 0.
- 8 bits de datos, primero el menos significativo.
- 1 bit de paridad (paridad impar).
- 1 bit de parada. Siempre 1.
- 1 bit de reconocimiento (Desde el PC al dispositivo, únicamente)

Los datos enviados desde el dispositivo son leídos en el flanco de bajada de la señal de reloj. Los datos enviados desde el PC al dispositivo son leídos en el flanco de subida. La frecuencia del reloj debe estar entre 10 y 16.7 kHz.



# Comunicación

## Dispositivo-PC

Cuando el ratón quiere enviar información, primero comprueba que la línea de reloj esta alta. Si no lo está, el PC esta inhibiendo la comunicación y por tanto el dispositivo debe almacenar en un buffer cualquier dato que quiera enviar hasta que el PC libere el reloj. La señal de reloj debe estar activa durante al menos 50 microsegundos para que el dispositivo empiece la transmisión.

## PC-Dispositivo

Existen algunas diferencias en este caso. Si el PC quiere enviar datos, primero tiene que poner las señales de reloj y datos en el estado Request-to-Send, inhibiendo primero la señal de reloj (baja) durante 100 microsegundos.

Cuando el dispositivo detecta este estado, empieza a generar señales de reloj y datos (8 bits y uno de parada). El PC cambia la línea de datos solo cuando la señal de reloj esta baja y el dato es leído cuando la señal esta alta, justo al contrario de lo que ocurría en la comunicación Dispositivo-PC.



Los pasos que el PC debe seguir son:

1. Colocar la línea de reloj baja durante al menos 100 microsegundos.
2. Bajar la línea de datos.
3. Liberar la línea de reloj.
4. Esperar a que el dispositivo baje la línea de reloj.
5. Poner a 1 la línea de datos para enviar el primer bit.
6. Esperar a que el dispositivo active la línea de reloj.
7. Esperar a que el dispositivo baje la línea de reloj.
8. Repetir los pasos 5-7 para los otros 7 bits de datos y el bit de paridad.
9. Liberar la línea de datos.
10. Esperar a que el dispositivo baje la línea de datos.
11. Esperar a que el dispositivo baje la línea de Reloj.
12. Esperar a que el dispositivo libere la línea de datos y de reloj.



## Entradas, Resolución y Escalado

La interfaz PS/2 estándar soporta las entradas Movimiento X (derecha/izquierda), Movimiento Y (arriba/abajo), Botón izquierdo, Botón central y Botón derecho. El ratón lee estas entradas con una frecuencia regular y actualiza varios contadores y flags para reflejar el movimiento y el estado de los botones.

El ratón estándar tiene dos **contadores** que llevan cuenta del movimiento: el contador de **movimiento X y el Y**. Son dos valores de 9 bits en complemento a dos con un flag de desbordamiento asociado cada uno. Su contenido, junto con el estado de los tres botones, es enviado al PC como un pack de tres bytes. Los contadores representan la cantidad de movimiento que se ha producido desde el último paquete de movimiento que fue enviado al PC, no representan posiciones absolutas.

Cuando el ratón lee sus entradas, graba el estado actual de los botones y comprueba si hay movimiento. Si se ha producido incrementa o decrementa los contadores X e Y. Si alguno de los contadores se desborda, pone a 1 el flag correspondiente.

El parámetro que determina la cantidad en que se debe incrementar o decrementar los contadores es la **resolución**. Por defecto es de 4 cuentas por milímetro y el PC puede cambiar el valor usando el comando “Set Resolution” (0xE8).

Hay otro parámetro que no afecta a los contadores de movimiento, pero sí al valor de estos. Este parámetro es la **escala**. Por defecto el ratón utiliza una escala 1:1, que no tiene efecto en el movimiento de ratón. Sin embargo, el PC debe seleccionar.



## Paquete de datos de movimiento

Como vimos, el ratón envía la información de movimiento y de estado de los botones usando el siguiente paquete de 3 bytes.

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 1	Y overflow	X overflow	Y bit signo	X bit signo	Siempre 1	Btn Central	Btn Derecho	Btn Izquierdo
Byte 2	Movimiento X							
Byte 3	Movimiento Y							

Los contadores de movimiento son enteros en complemento a 2 de 9 bits (rango -255 a +255), donde el bit más significativo es el bit de signo que aparece en el byte 1 del paquete. Estos contadores son actualizados cuando el ratón lee sus entradas y ve que se ha producido un movimiento. El valor es la cantidad de movimiento que se ha producido desde el último envío de datos de movimiento. Después de cada envío, los contadores de movimiento son reseteados. También se resetean después de la recepción de cualquier comando desde el PC, salvo el de reenvío (0xFE).



## Modos de operación

El manejo de los datos se lleva a cabo según el modo en que el ratón este operando. Hay cuatro modos estándar de operación:

- *Reset* – El ratón entra en el modo reset en el encendido o después de recibir un comando Reset.
- *Stream* – Es el modo por defecto (después de que termine de ejecutarse el Reset) y es el modo en el que la mayoría del software utiliza el ratón. "Set Stream Mode" (0xEA).
- *Remoto* – El modo remoto es útil en algunas situaciones y puede ser accedido con el comando "Set Remote Mode" (0xF0).
- *Wrap* – Este modo no es demasiado útil salvo para testear la conexión entre el ratón y el PC. El modo Wrap se activa con el comando "Set Wrap Mode" (0xEE). Para salir de este modo se debe realizar un Reset.

Pueden existir otros modos de operación extendidos, pero no pertenecen al estándar PS/2.



### Modo Reset

Al entrar en este modo, el ratón realiza un *autotesteo* y establece los siguientes *valores por defecto*:

- Tasa de muestreo = 100 muestras/seg
- Resolución = 4 cuentas/mm
- Escalado = 1:1
- Notificación de datos desactivada.

### Modo Stream

En este modo, el ratón envía datos de movimiento cuando detecta movimiento o cambia el estado de alguno de los botones. La tasa máxima a la cual puede enviar datos es el Sample rate. Este parámetro esta en el rango desde 10 muestras/segundo a 200 muestras/seg. El valor predeterminado es de 100 y el PC puede cambiar este valor usando el comando "Set Sample Rate".

### Modo Remoto

En este modo el ratón lee sus entradas y actualiza los contadores y flags a la tasa de muestreo actual, pero no envía automáticamente los paquetes de datos cuando ocurre un movimiento; en este caso es el PC que solicita el paquete con el comando "Read Data". Al recibir este comando, el ratón envía un paquete de movimiento y resetea sus contadores.

### Modo Wrap

Este es un modo "eco" en el que cada byte recibido por el ratón es reenviado hacia el PC. Aunque el byte sea un comando invalido el ratón no responderá a él, salvo al comando "Reset" y el "Reset Wrap Mode".





## Extensiones Intellimouse

Una extensión popular del estándar PS/2 es *Intellimouse de Microsoft*. Soporta un total de 5 botones y tres tipos de movimientos, a los que ha añadido la rueda central de reemplazamiento. El paquete de datos es de 4 bytes.

Como el PS/2 no reconoce este paquete de datos, el ratón opera como un estándar a menos que sepa que los drivers soportan este formato extendido, es decir, los botones 4 y 5 y la rueda de desplazamiento no funcionarán. Para activar la rueda de desplazamiento el PC envía los comandos para cambiar la tasa de muestreo a 200, a 100 y a 80 sucesivamente. Tras esto envía el comando "Get device ID" y si es un ratón PS/2 responderá con un identificador 00h. Si es un Intellimouse responderá con un 03h. El formato de 4 bytes es el siguiente:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 1	Y overflow	X overflow	Y bit signo	X bit signo	Siempre 1	Botón Central	Botón Derecho	Botón Izquierdo
Byte 2	Movimiento X							
Byte 3	Movimiento Y							
Byte 4	Movimiento Z							

El movimiento Z es un número en complemento a 2 que representa el movimiento de la rueda de desplazamiento desde el último envío de datos. El rango de valores es de -8 a 7, lo que significa que solo se utilizan los 4 bits menos significativos y los 4 más significativos son la extensión del signo.



Para utilizar el modo "Rueda de desplazamiento + botón 5", el PC cambia la tasa de muestreo a 200 dos veces y después a 80. El ratón entonces responde con un ID 04h. El formato de este paquete de datos es:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 1	Y overflow	X overflow	Y bit signo	X bit signo	Siempre 1	Btn Central	Btn Der	Btn Izq
Byte 2	Movimiento X							
Byte 3	Movimiento Y							
Byte 4	Siempre 0	Siempre 0	Btn 5	Btn 4	Z3	Z2	Z1	Z0

**Z0-Z3** es un número en complemento que representa el movimiento de la rueda de desplazamiento desde el último envío de datos. El rango es de nuevo de -8 a +7.

**Btn 4:** A 1 si el botón 4 es presionado, a 0 si no.

**Btn 5:** A 1 si el botón 5 es presionado, a 0 si no.



## Comandos

Este es el conjunto de comandos del estándar PS/2 del ratón. Si éste está en modo Stream, el PC debe deshabilitar el envío de datos desde el ratón (comando F5h) antes de enviar cualquier otra orden.

- FFh (Reset) – El Ratón responde a este comando con un comando de reconocimiento (FAh) y entra en el modo Reset.
- FEh (Reenvio) – El PC envía este comando cada vez que recibe un dato invalido del ratón. El ratón reenvía el último paquete.
- F6h (Cargar Predeterminados) – El ratón responde con esta señal de reconocimiento y después carga los valores: Tasa de muestreo=100, Resolución=4 cuentas/seg, Escalado=1:1 y deshabilita el envío de datos. Entonces entra en modo Stream después de resetear los contadores.
- F5h (Deshabilitar envío de datos) – El ratón responde con la señal de reconocimiento, desactiva el envío de datos y resetea los contadores.
- F4h (Activa el envío de datos) – El ratón responde con la señal de reconocimiento, activa el envío de datos y resetea los contadores.

17



Alberto Pérez Pérez | Periféricos, Interfaces y Buses.

- F3h (Establece la tasa de muestreo) – El ratón responde con la señal de reconocimiento y lee un nuevo byte del PC. Este será la nueva tasa de muestreo. Tasas validas son 10, 20, 40, 60, 80, 100 y 200.
- F2h (Obtener ID) - El ratón responde con la señal de reconocimiento y envía el byte de identificación. También resetea sus contadores.
- F0h (Modo Reset) - El ratón responde con la señal de reconocimiento, después resetea sus contadores y entra en modo Reset.
- EEh (Modo Wrap) - El ratón responde con la señal de reconocimiento, después resetea sus contadores y entra en modo Wrap.
- ECh (Modo Warp Reset) – El ratón responde con la señal de reconocimiento, después resetea sus contadores y entra en el modo anterior al Wrap (Stream o Remoto).
- EBh (Leer Datos) – El ratón responde con la señal de reconocimiento y envía el paquete de datos. Es la única forma de leer datos en el modo Remoto. Si el envío de datos ha sido correcto, resetea sus contadores.
- EAh (Modo Stream) – El ratón responde con la señal de reconocimiento, después resetea sus contadores y entra en modo Stream.

18



Alberto Pérez Pérez | Periféricos, Interfaces y Buses.

- E9h (Envío de Estado) – El ratón responde con la señal de reconocimiento y envía el siguiente paquete de 3 bytes de estado. Después resetea los contadores.

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 1	Siempre 0	Modo	Enable	Escala	Siempre 0	Btn Izq	Btn Cnt	Btn Der
Byte 2	Resolución							
Byte 3	Tasa de muestreo							

*Botones Derecho, Central y Izquierdo* = 1 si el botón es presionado.

*Escala* = 1 si la escala es 2:1; 0 si es 1:1.

*Enable* = 1 si el envío de datos esta activado.

*Modo* = 1 si el Modo Remoto esta activado.

- E8h (Establece Resolución) - El ratón responde con la señal de reconocimiento, lee el byte enviado por el PC y envía de nuevo la señal de reconocimiento, reseteando los contadores finalmente. El byte leído establece la resolución:

Byte	Resolucion
0x00	1 cuentas/mm
0x01	2 cuentas /mm
0x02	4 cuentas /mm
0x03	8 cuentas /mm



- E7h (Establecer Escala 2:1) - El ratón responde con la señal de reconocimiento y activa la escala 2:1.
- E6h (Establecer Escala 1:1) - El ratón responde con la señal de reconocimiento y activa la escala 1:1.

Los únicos comandos que puede enviar el ratón estándar son "Reenvío" (FEh) y "Error" (FCh). Ambos funcionan de la misma manera que los comandos PC-dispositivo.

## Inicialización

El ratón PS/2 es inicializado solo cuando el PC se inicia, por lo que no posee conexión en caliente y será necesario reiniciar el sistema para añadir o quitarlo (además, algunas placas podrían dañarse si se realiza la conexión/desconexión en caliente).

La detección del ratón se realiza durante el arranque y la BIOS permite al sistema operativo configurar y habilitarlo. Si no se detecta el dispositivo la BIOS inhibe la comunicación de ese bus. Si se arranca el sistema con el ratón conectado, el sistema operativo Windows puede que lo detecte. Microsoft intento dar soporte a esto, pero solo funciona el 50% de las veces.



## Bibliografía

- <http://www.computer-engineering.org/ps2protocol/>
- <http://www.computer-engineering.org/ps2mouse/>
- <http://www.tavi.co.uk/ps2pages/ohland/mouse.html>
- Wikipedia.



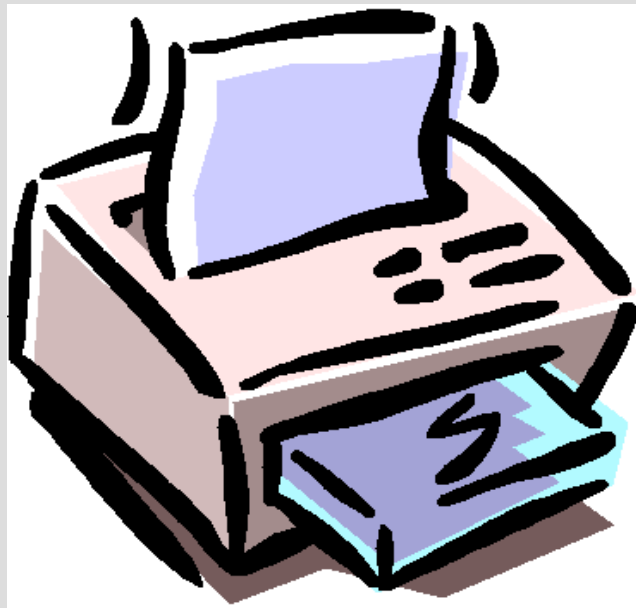
# Dispositivos de E/S de datos



## Bloque II (actividad desarrollada por los alumnos)

- ratón
- **impresoras**
- monitor táctil

# LA IMPRESORA



PERIFÉRICOS, INTERFACES Y BUSES  
Javier de la Dehesa Cueto-Felgueroso

## INTRODUCCIÓN

- Plasman información digital en un soporte físico (p.e. papel)
- Periférico local (USB, puerto paralelo) o de red (ethernet)
- Integrado con otros dispositivos (escáner, fax, etc)
- Funcionamiento electromecánico y por lo tanto lento

# TIPOS DE IMPRESORAS

- Impresoras láser
  - Las más comunes en empresas, pueden ser a blanco y negro o a color
  - Se basa en un tambor fotorreceptor sensible a la luz
  - Un láser carga las zonas adecuadas del tambor formando una imagen electrostática
  - El tambor es rociado con un polvo negro cargado (toner) que es repelido por las zonas cargadas quedando plasmado en el papel en esos puntos

# TIPOS DE IMPRESORAS

- Impresoras de inyección de tinta
  - Dominan el mercado doméstico
  - Desplaza lateralmente unos cabezales a través del papel imprimiendo líneas
  - Obtiene la tinta de cartuchos reemplazables
  - Tienen un coste inicial bajo, pero los recambios son caros (modelo de negocio)

# TIPOS DE IMPRESORAS

- Impresoras de impacto
  - Imprimen por opresión de una cinta sobre el papel (como una máquina de escribir)
  - Pueden ser
    - **De tipos:** impactan con tipos contenidos en la impresora
    - **De matriz de puntos:** hacen pequeños impactos de acuerdo a una matriz de puntos

# TIPOS DE IMPRESORAS

- Impresoras de tinta sólida
  - Emplean bloques de tinta sólida que se derriten y se aplican sobre el medio
  - Utilizado para imprimir a color en empresas, pues imprime sobre transparencias y otros medios
  - Dan una buena calidad, respetan el medio ambiente y son fáciles de usar
  - Requieren calentamiento previo y son frágiles

# TIPOS DE IMPRESORAS

- Otros tipos:
  - **Impresoras de sublimación:** funcionan a base de calor, se emplean para color de alta calidad
  - **Plotter:** para imprimir gráficos vectoriales
  - **Impresoras UV:** imprimen con luz ultravioleta sobre papel especial, es una tecnología en desarrollo
  - **Impresoras de propósito especial**

# HARDWARE

- La impresora se ha controlado tradicionalmente desde el puerto paralelo (LPTx)
- Las direcciones de I/O del mismo pueden variar, aunque normalmente se mapea el puerto LPT1 a 0x378 y LPT2 a 0x278
- Puede emplear interrupciones, normalmente a través de la línea IRQ7



# HARDWARE

- Al margen del mecanismo de impresión, el controlador de una impresora dispone de los siguientes registros
  - **Registro de datos:** sólo escritura, por el que se envían los caracteres a imprimir
  - **Registro de estado:** sólo lectura, informa del estado actual de la impresora
  - **Registro de control:** sólo escritura, sirve para enviar comandos al periférico

## REGISTRO DE DATOS

0	1	2	3	4	5	6	7
CHR0	CHR1	CHR2	CHR3	CHR4	CHR5	CHR6	CHR7

- Almacena el carácter a enviar a la impresora
- El carácter se escribe en el registro y posteriormente se da la orden de envío

# REGISTRO DE ESTADO

0	1	2	3	4	5	6	7
-	-	-	$\neg$ ERROR	SLCT	PE	$\neg$ ACK	$\neg$ BUSY

- $\neg$ ERROR: Activo si hay un error grave
- SLCT: La impresora está activa (*online*)
- PE: No hay papel (*paper error*)
- $\neg$ ACK: Confirmación de recepción de carácter
- $\neg$ BUSY: Impresora ocupada

# REGISTRO DE CONTROL

0	1	2	3	4	5	6	7
$\neg$ STROBE	$\neg$ AUTOFEED	$\neg$ INIT	SLCTIN	IRQENABLE	-	-	-

- $\neg$ STROBE: Envío de carácter
- $\neg$ AUTOFEED: Salto del línea con carácter 13 (CR)
- $\neg$ INIT: Reiniciar la impresora
- SLCTIN: Selecciona la impresora (0 para *offline*)
- IRQENABLE: Habilita las interrupciones

# PROGRAMACIÓN

- El esquema de envío de información a la impresora es el siguiente:
  - Escribir el carácter en el registro de datos
  - Activar de STROBE
  - La impresora activa BUSY
  - Esperar a que se active ACK
  - Desactivar STROBE
  - Repetir con el resto de caracteres

# PROGRAMACIÓN

- Puede mejorarse la eficiencia utilizando interrupciones (activando IRQENABLE)
- La impresora genera una interrupción cada vez que recibe un carácter
- En vez de esperar a que se active ACK (por *polling*), el driver se suspende hasta que llega la interrupción
- El primer carácter se envía “manualmente” y los siguientes a través de la rutina de atención a la interrupción

# PROGRAMACIÓN

- Consideraciones especiales:
  - Se debe desactivar STROBE una vez se reciba el carácter para evitar leerlo dos veces
  - El uso de AUTOFEED cambia en función del sistema operativo:
    - Activo para Unix (CR)
    - Inactivo para Windows (CRLF) y Mac (LF)
  - Es necesario comprobar errores tras cada envío (PE, ERROR)

## Dispositivos de E/S de datos



### Bloque II (actividad desarrollada por los alumnos)

- ratón
- impresoras
- **monitor táctil**

# Pantalla Táctil

- Es un dispositivo muy de moda actualmente.
- Presente en gran cantidad de dispositivos. Muy utilizada en móviles y PDAs.

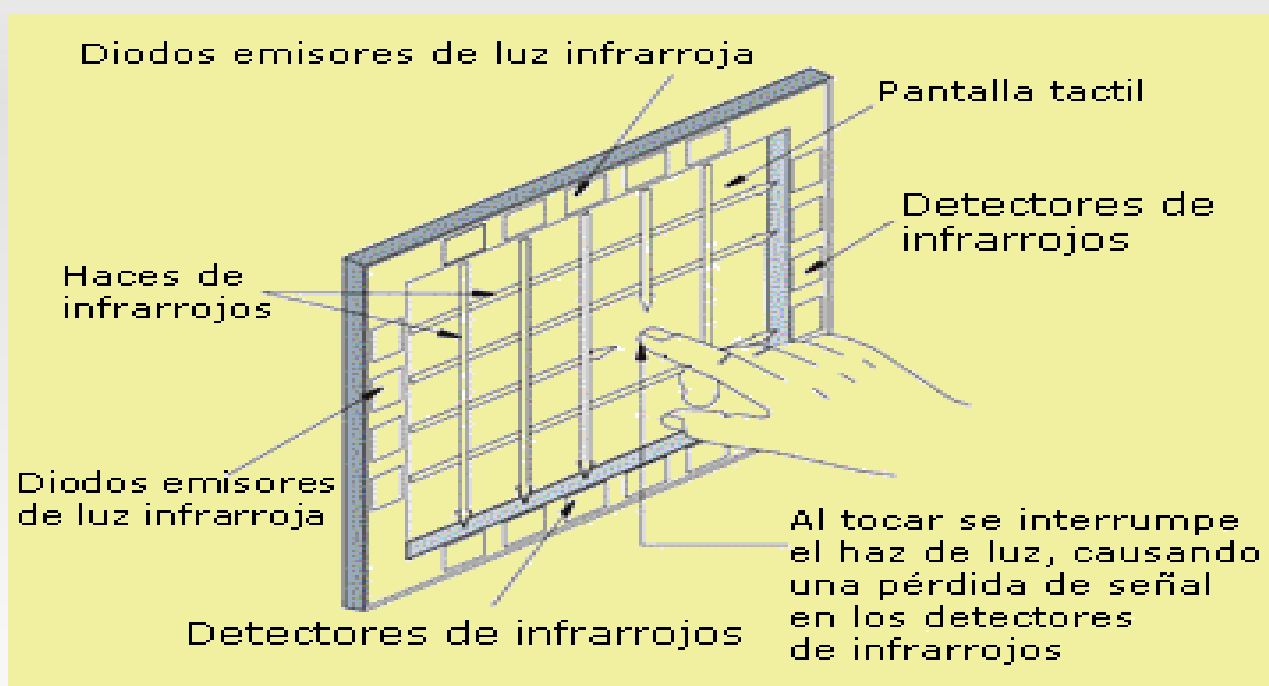
# Pantalla Táctil

- Existen diversas tecnologías:
  - Pantallas táctiles por infrarrojos
  - Pantallas táctiles resistivas
  - Pantallas táctiles capacitivas
  - Pantallas táctiles de onda acústica superficial, (SAW)

# Pantallas táctiles por infrarrojos

- Es el sistema más antiguo y fácil de entender:
- En los bordes de la pantalla, en la carcasa de la misma, existen unos emisores y receptores de infrarrojos. En un lado de la pantalla están los emisores y en el contrario los receptores. Tenemos una matriz de rayos infrarrojos vertical y horizontal. Al pulsar con el dedo o con cualquier objeto, sobre la pantalla interrumpimos un haz infrarrojo vertical y otro horizontal.

# Pantallas táctiles por infrarrojos



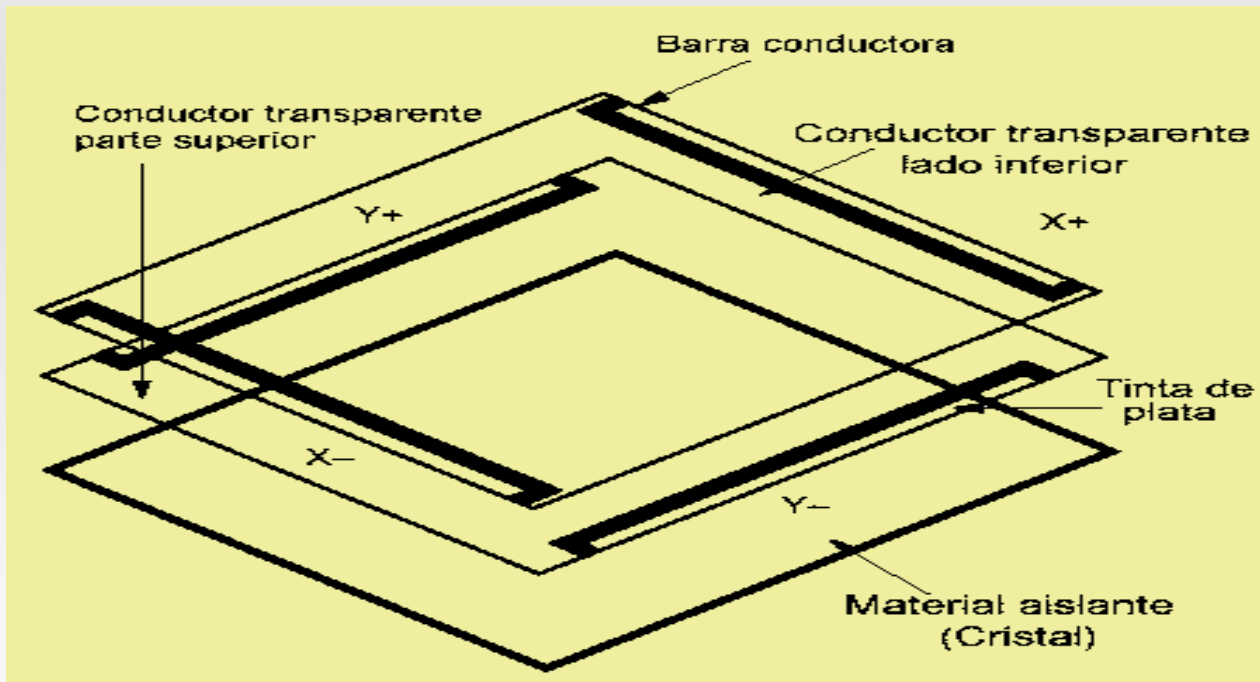
# Pantallas táctiles por infrarrojos

- Ventajas:
- Simplicidad
- No se oscurece la pantalla
- Desventajas:
- Caras
- Voluminosas
- Muy sensibles a la suciedad y a las falsas pulsaciones.

# Pantallas táctiles resistivas

- Es un tipo de pantallas táctiles muy usado
- La pantalla táctil propiamente dicha está formada por dos capas de material conductor transparente, con una cierta resistencia a la corriente eléctrica, y con una separación entre las dos capas. Cuando se toca la capa exterior se produce un contacto entre las dos capas conductoras. Un sistema electrónico detecta el contacto y midiendo la resistencia puede calcular el punto de contacto.

# Pantallas táctiles resistivas



# Pantallas táctiles resistivas

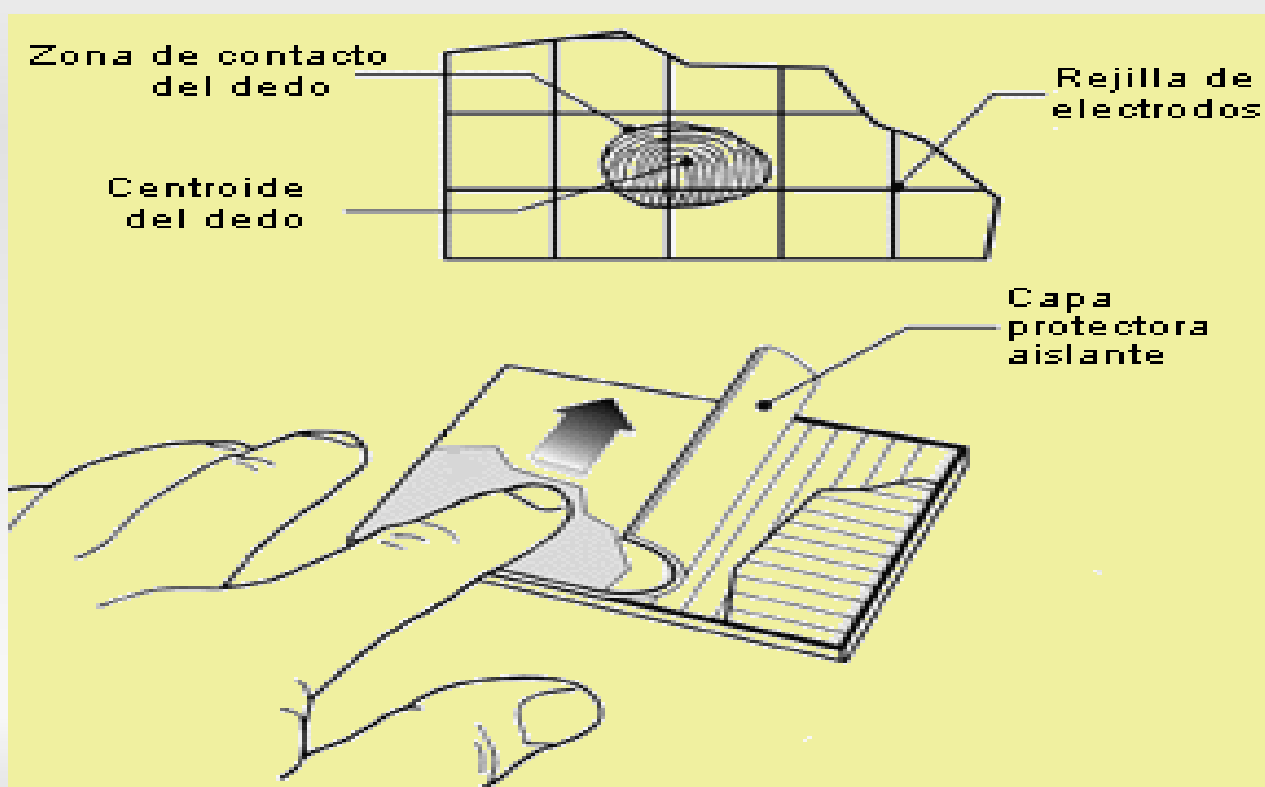
- Ventajas:
  - Pueden ser usadas con cualquier objeto, un dedo, un lápiz, un dedo con guantes, etc.
  - Son económicas, fiables y versátiles
- Desventajas:
  - Al usar varias capas de material transparente sobre la propia pantalla, se pierde bastante luminosidad.
  - Sensible a la luz ultravioleta, de tal forma que con el tiempo se degrada y pierde flexibilidad y transparencia



# Pantallas táctiles capacitivas

- En estas pantallas se añade una capa conductora al cristal del propio tubo.
- Se aplica una tensión en cada una de las cuatro esquinas de la pantalla.
- Una capa que almacena cargas se sitúa sobre el cristal del monitor.
- Cuando un usuario toca el monitor algunas cargas se transfieren al usuario, de tal forma que la carga en la capa capacitiva se decrementa.
- Este decrecimiento se mide en los circuitos situados en cada esquina del monitor.
- Se calcula, por la diferencia de carga entre cada esquina, el sitio concreto donde se tocó.

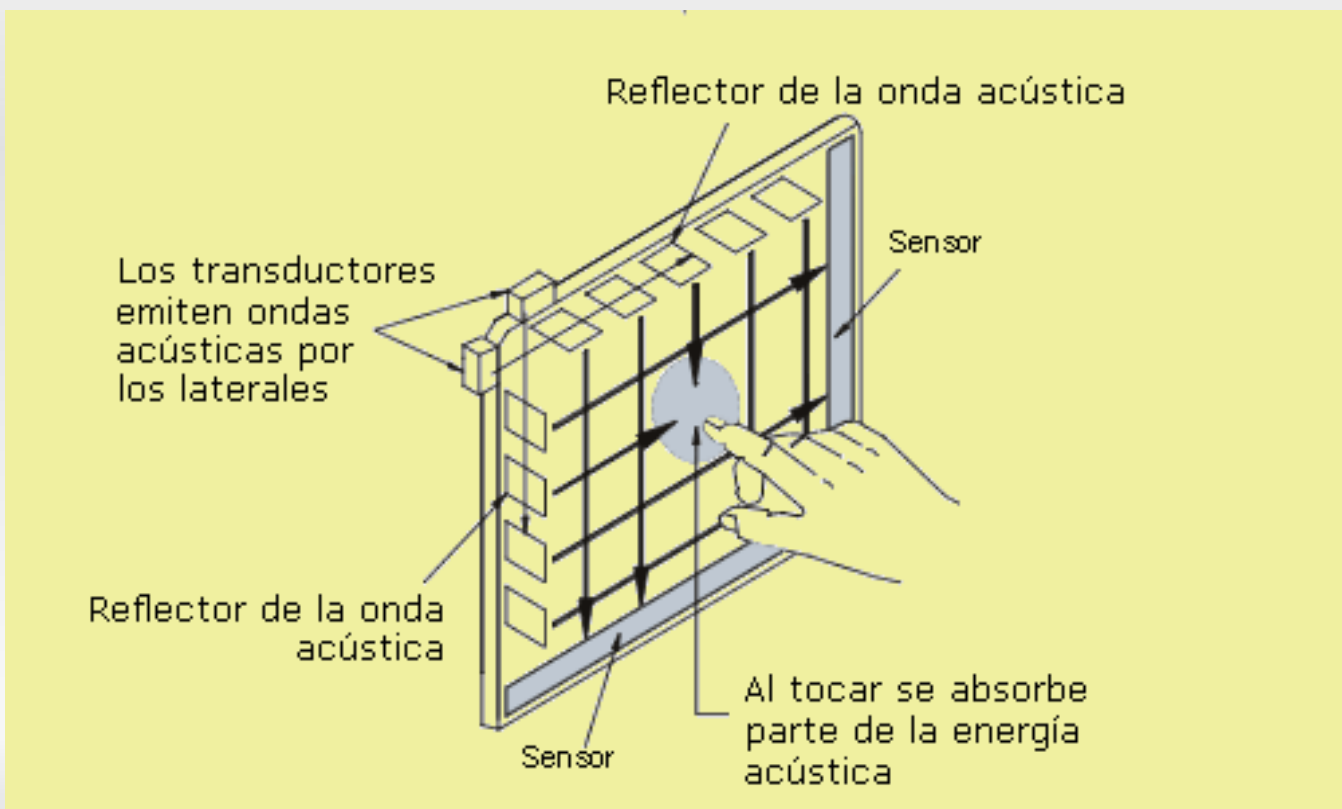
# Pantallas táctiles capacitivas



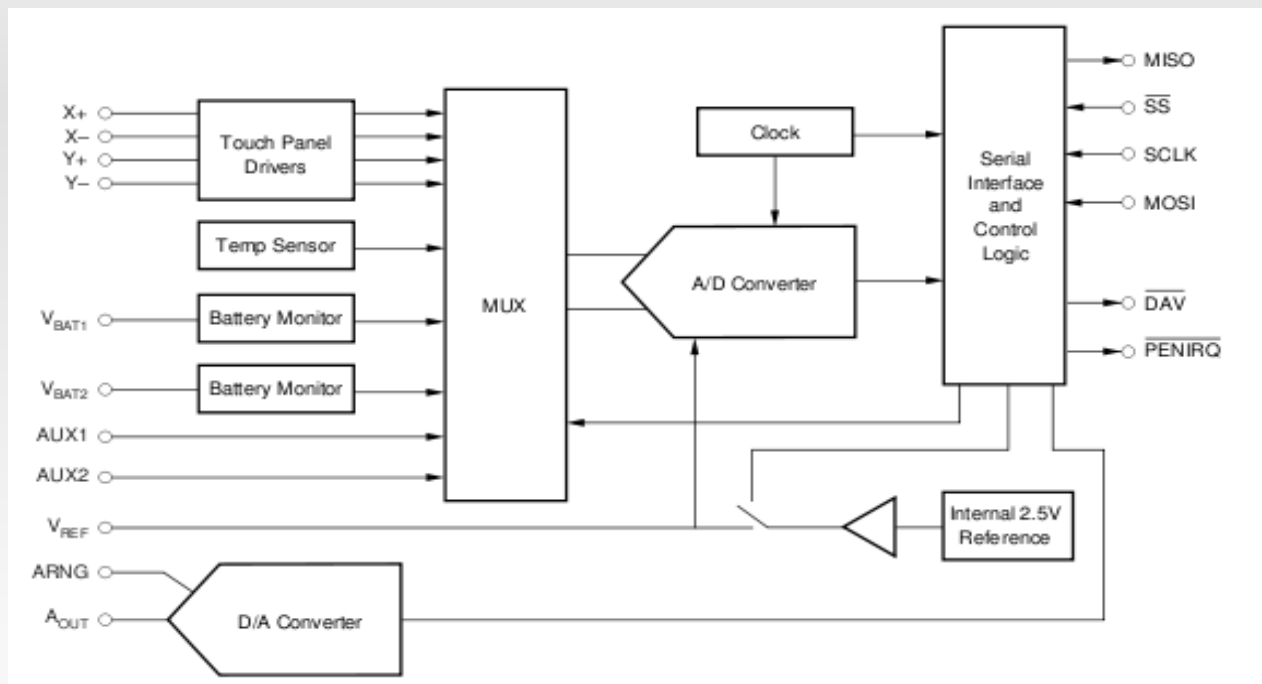
# Pantallas táctiles de onda acústica superficial, (SAW)

- A través de la superficie del cristal se transmiten dos ondas acústicas inaudibles para el hombre. Una de las ondas se transmite horizontalmente y la otra verticalmente. Cada onda se dispersa por la superficie de la pantalla rebotando en unos reflectores acústicos.
- Las ondas acústicas no se transmiten de forma continua, sino por trenes de impulsos. Dos detectores reciben las ondas, uno por cada eje. Se conoce el tiempo de propagación de cada onda acústica en cada trayecto. Cuando el usuario toca con su dedo en la superficie de la pantalla, el dedo absorbe una parte de la potencia acústica, atenuando la energía de la onda.

# Pantallas táctiles de onda acústica superficial, (SAW)

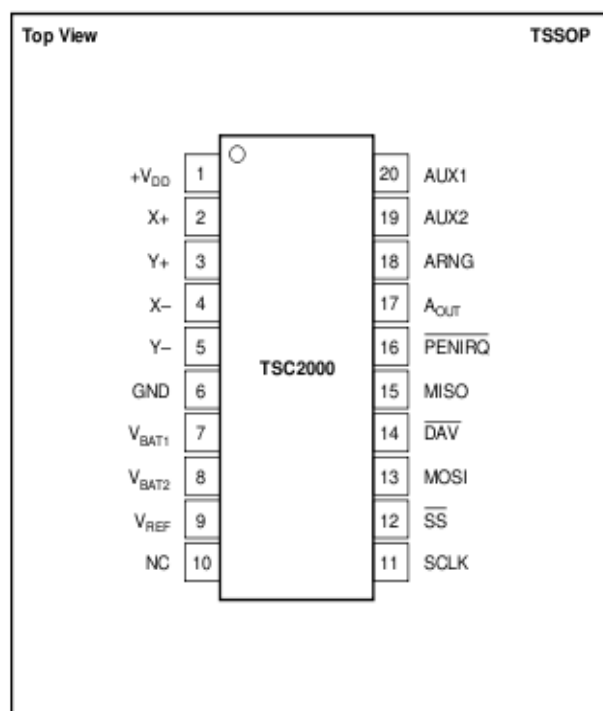


# TSC2000 circuito controlador de pantallas táctiles resistivas



# TSC2000 circuito controlador de pantallas táctiles resistivas

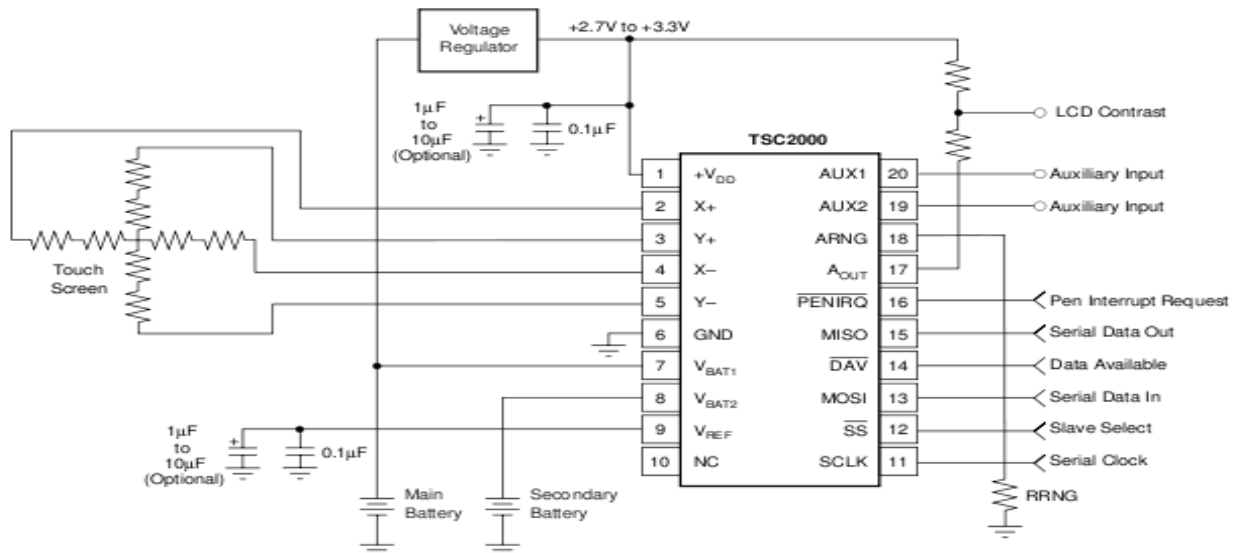
## PIN CONFIGURATION



## PIN DESCRIPTION

PIN	NAME	DESCRIPTION
1	V <sub>DD</sub>	Power Supply
2	X+	X+ Position Input
3	Y+	Y+ Position Input
4	X-	X- Position Input
5	Y-	Y- Position Input
6	GND	Ground
7	V <sub>BAT1</sub>	Battery Monitor Input 1
8	V <sub>BAT2</sub>	Battery Monitor Input 2
9	V <sub>REF</sub>	Voltage Reference Input/Output
10	NC	No Connection
11	SCLK	Serial Clock Input
12	SS	Slave Select Input (Active LOW). Data will not be clocked in to MOSI unless SS is LOW. When SS is HIGH, MISO is high impedance.
13	MOSI	Serial Data Input. Data is clocked in at SCLK falling edge.
14	DAV	Data Available (Active LOW)
15	MISO	Serial Data Output. Data is clocked out at SCLK falling edge. High impedance when SS is HIGH.
16	PENIRQ	Pen Interrupt
17	A <sub>OUT</sub>	Analog Output Current from D/A Converter
18	ARNG	D/A Converter Analog Output Range Set
19	AUX2	Auxiliary A/D Converter Input 2
20	AUX1	Auxiliary A/D Converter Input 1

# TSC2000 circuito controlador de pantallas táctiles resistivas



# TSC2000 circuito controlador de pantallas táctiles resistivas

- TSC2000 utiliza para comunicarse un SPI bus.
- El SPI permite full-duplex, comunicación síncrona serie entre un host (el master) y un periférico(esclavo).
- El master genera el reloj síncrono y inicia la transmisión.
- El esclavo depende del master para iniciar una transmisión.

# TSC2000 circuito controlador de pantallas táctiles resistivas

PG3	PG2	PG1	PG0	PAGE ADDRESSED
0	0	0	0	0
0	0	0	1	1
0	0	1	0	Reserved
0	0	1	1	Reserved
0	1	0	0	Reserved
0	1	0	1	Reserved
0	1	1	0	Reserved
0	1	1	1	Reserved
1	0	0	0	Reserved
1	0	0	1	Reserved
1	0	1	0	Reserved
1	0	1	1	Reserved
1	1	0	0	Reserved
1	1	0	1	Reserved
1	1	1	0	Reserved
1	1	1	1	Reserved

MSB BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	LSB BIT 0
R/W	PG3	PG2	PG1	PG0	ADDR5	ADDR4	ADDR3	ADDR2	ADDR1	ADDR0	X	X	X	X	X

# TSC2000 circuito controlador de pantallas táctiles resistivas

PAGE 0: DATA REGISTERS		PAGE 1: CONTROL REGISTERS	
ADDR	REGISTER	ADDR	REGISTER
00	X	00	ADC
01	Y	01	Reserved
02	Z <sub>1</sub>	02	DACCTL
03	Z <sub>2</sub>	03	REF
04	Reserved	04	RESET
05	BAT1	05	CONFIG
06	BAT2	06	Reserved
07	AUX1	07	Reserved
08	AUX2	08	Reserved
09	TEMP1	09	Reserved
0A	TEMP2	0A	Reserved
0B	DAC	0B	Reserved
0C	Reserved	0C	Reserved
0D	Reserved	0D	Reserved
0E	Reserved	0E	Reserved
0F	Reserved	0F	Reserved
10	ZERO	10	Reserved
11	Reserved	11	Reserved
12	Reserved	12	Reserved
13	Reserved	13	Reserved
14	Reserved	14	Reserved
15	Reserved	15	Reserved
16	Reserved	16	Reserved
17	Reserved	17	Reserved
18	Reserved	18	Reserved
19	Reserved	19	Reserved
1A	Reserved	1A	Reserved
1B	Reserved	1B	Reserved
1C	Reserved	1C	Reserved
1D	Reserved	1D	Reserved
1E	Reserved	1E	Reserved
1F	Reserved	1F	Reserved

# TSC2000 A/D CONVERTER CONTROL REGISTER (PAGE 1, ADDRESS 00H)

MSB BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	LSB BIT 0
PSM	STS	AD3	AD2	AD1	AD0	RS1	RS0	AV1	AV0	CL1	CL0	PV2	PV1	PV0	X

PSM		
READ/WRITE	VALUE	DESCRIPTION
Read	0	No Screen Touch Detected
Read	1	Screen Touch Detected
Write	0	Conversions Controlled by Host
Write	1	Conversions Controlled by TSC2000

STS		
READ/WRITE	VALUE	DESCRIPTION
Read	0	Converter is Busy
Read	1	Conversions are Complete, Data is Available
Write	0	Normal Operation
Write	1	Stop Conversion and Power Down

RS1	RS0	FUNCTION
0	0	12-Bit Resolution. Power up and reset default.
0	1	8-Bit Resolution
1	0	10-Bit Resolution
1	1	12-Bit Resolution

# TSC2000 A/D CONVERTER CONTROL REGISTER (PAGE 1, ADDRESS 00H)

A/D3	A/D2	A/D1	A/D0	FUNCTION
0	0	0	0	Invalid. No registers will be updated. This is the default state after a reset.
0	0	0	1	Touch screen scan function: X and Y coordinates converted and the results returned to X and Y data registers. Scan continues until either the pen is lifted or a stop bit is sent.
0	0	1	0	Touch screen scan function: X, Y, Z <sub>1</sub> , and Z <sub>2</sub> coordinates converted and the results returned to X, Y, Z <sub>1</sub> , and Z <sub>2</sub> data registers. Scan continues until either the pen is lifted or a stop bit is sent.
0	0	1	1	Touch screen scan function: X coordinate converted and the results returned to X data register.
0	1	0	0	Touch screen scan function: Y coordinate converted and the results returned to Y data register.
0	1	0	1	Touch screen scan function: Z <sub>1</sub> and Z <sub>2</sub> coordinates converted and the results returned to Z <sub>1</sub> and Z <sub>2</sub> data registers.
0	1	1	0	Battery Input 1 converted and the results returned to the BAT1 data register.
0	1	1	1	Battery Input 2 converted and the results returned to the BAT2 data register.
1	0	0	0	Auxiliary Input 1 converted and the results returned to the AUX1 data register.
1	0	0	1	Auxiliary Input 2 converted and the results returned to the AUX2 data register.
1	0	1	0	A temperature measurement is made and the results returned to the temperature measurement 1 data register.
1	0	1	1	Port scan function: Battery Input 1, Battery Input 2, Auxiliary Input 1, and a Auxiliary Input measurements are made and the results returned to the appropriate data registers.
1	1	0	0	A differential temperature measurement is made and the results returned to the temperature measurement 2 data register.
1	1	0	1	Turn on X+, X- drivers.
1	1	1	0	Turn on Y+, Y- drivers.
1	1	1	1	Turn on Y+, X- drivers.

# TSC2000 A/D CONVERTER CONTROL REGISTER (PAGE 1, ADDRESS 00H)

AV1	AV0	FUNCTION
0	0	None
0	1	4 Data Averages
1	0	8 Data Averages
1	1	16 Data Averages

CL1	CL0	FUNCTION
0	0	8MHz Internal Clock Rate—8-Bit Resolution Only
0	1	4MHz Internal Clock Rate—10-Bit Resolution Only
1	0	2MHz Internal Clock Rate.
1	1	1MHz Internal Clock Rate.

PV2	PV1	PV0	FUNCTION
0	0	0	0 $\mu$ s Stabilization Time
0	0	1	100 $\mu$ s Stabilization Time
0	1	0	500 $\mu$ s Stabilization Time
0	1	1	1ms Stabilization Time
1	0	0	5ms Stabilization Time
1	0	1	10ms Stabilization Time
1	1	0	50ms Stabilization Time
1	1	1	100ms Stabilization Time

# TSC2000 A/D CONVERTER CONTROL REGISTER (PAGE 1, ADDRESS 02H)

MSB	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	LSB	BIT 0
DPD	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	

DPD	
VALUE	DESCRIPTION
0	D/A Converter is Powered and Operational
1	D/A Converter is Powered Down

# REFERENCE REGISTER (PAGE 1, ADDRESS 03H)

MSB															LSB
BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
X	X	X	X	X	X	X	X	X	X	X	INT	DL1	DL0	PDN	RFV

INT	
VALUE	DESCRIPTION
0	External Reference Selected
1	Internal Reference Selected

PDN	
VALUE	DESCRIPTION
0	Internal Reference is Powered at All Times
1	Internal Reference is Powered Down Between Conversions

INT	PDN	REFERENCE BEHAVIOR
0	0	External Reference Used, Internal Reference Powered Down
0	1	External Reference Used, Internal Reference Powered Down
1	0	Internal Reference Used, Always Powered Up
1	1	Internal Reference Used, Will Power Up During Conversions and Then Power Down

RFV	
VALUE	DESCRIPTION
0	1.25V Reference Voltage
1	2.5V Reference Voltage

# TSC2000 CONFIGURATION CONTROL REGISTER(PAGE 1, ADDRESS 05H)

MSB															LSB
BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
X	X	X	X	X	X	X	X	X	X	PRE2	PRE1	PRE0	SNS2	SNS1	SNS0

PRE[2:0]			
PRE2	PRE1	PRE0	TIME
0	0	0	20 $\mu$ s
0	0	1	84 $\mu$ s
0	1	0	276 $\mu$ s
0	1	1	340 $\mu$ s
1	0	0	1.044ms
1	0	1	1.108ms
1	1	0	1.300ms
1	1	1	1.364ms

SNS[2:0]			
SNS2	SNS1	SNS0	TIME
0	0	0	32 $\mu$ s
0	0	1	96 $\mu$ s
0	1	0	544 $\mu$ s
0	1	1	608 $\mu$ s
1	0	0	2.080ms
1	0	1	2.144ms
1	1	0	2.592ms
1	1	1	2.656ms