

120-22

A nuestros padres

Por habernos apoyado siempre,

A nuestros profesores

Por haber creído en nosotros

A nuestros amigos

Por haber estado ahí siempre

Gracias

Project 0

TK7887-5

B42

✓

61199535x



TECNOLÓGICO
DE MONTERREY.

Campus Ciudad de México

Instituto Tecnológico y de Estudios Superiores de Monterrey
Campus Ciudad de México
División de Ingeniería y Arquitectura

Ingeniería en Sistemas Electrónicos

Proyectos de Ingeniería

MOUSE PARA PERSONAS CON POCA O NULA MOVILIDAD EN LAS MANOS.

Profesor:

Dr. Martín Rogelio Bustamante Bello

Asesores:

Dr. Martín Rogelio Bustamante Bello

Firma Martín Rogelio Bustamante

Dr. José Ramón Álvarez Bada

Firma José Ramón Álvarez Bada

Agosto – Diciembre de 2008

Autores:

Rebeca Zoraya Becerril Fernández

A00969997

ISE

Firma Rebeca Becerril

Carlos Iván Monroy Badillo

A01109187

ISE

Firma Carlos Iván Monroy Badillo

CENTRO DE INVESTIGACIÓN EN MICROSYSTEMAS Y BIODISEÑO



INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE MONTERREY
CAMPUS CIUDAD DE MEXICO
DIVISION DE INGENIERIA Y ARQUITECTURA



TECNOLÓGICO
DE MONTERREY.

BIBLIOTECA
Campus Ciudad de México



Índice

1.	Introducción	6
1.1	Justificación	6
1.2	Problemática.....	6
1.3	Objetivos y propuestas de valor.....	8
1.3.1	Objetivos generales.....	8
1.3.2	Propuesta de valor.....	8
1.4	Alcances y limitaciones.....	8
1.4.1	Alcances	8
1.4.2	Limitaciones	9
1.5	Antecedentes.....	9
1.6	Base teórica.....	9
1.6.1	Discapacidad.....	9
1.6.2	Discapacidad en México.....	12
1.6.3	Discapacidad Motriz	12
1.6.4	Dispositivos de Interfaz Humana.....	13
1.6.5	Funcionamiento del <i>mouse</i>	15
1.6.6	Protocolo USB.....	23
1.6.7	USB 2.0	27
1.6.8	Microsistemas Electromecánicos (MEMS)	28
1.6.9	Microcontrolador ATMEGA128	34
2.	Propuesta de solución.....	38
2.1	Módulos del sistema.....	38
2.1.1	Acelerómetros.....	38
2.1.2	Medición de la inclinación con acelerómetros.....	38
2.1.3	Procesamiento (Microcontrolador)	44
2.1.4	Programación.....	45
2.1.5	<i>Driver mouse</i> USB	47
2.2	Integración del sistema	48
2.2.1	Interconexión del sistema.....	48
2.2.2	Diseño de la cubierta del prototipo funcional	50
2.2.3	Costo del prototipo experimental.....	52
2.2.4	Costos de un prototipo precomercial	53
3.	Resultados y Pruebas.....	55
3.1	Ingeniería inversa y adaptación del <i>driver</i> USB	55
3.2	Caracterización del acelerómetro como inclinómetro.....	58
3.3	Caracterización del Convertidor Analógico Digital.....	64
3.4	Pruebas de longitud del cable	67
3.5	Prueba de sensibilidad del desplazamiento	68
3.6	Problemas encontrados en el desarrollo del dispositivo.....	70
4.	Conclusiones y trabajo futuro.....	72
4.1	Conclusiones.....	72
4.2	Trabajo a futuro.....	74
5.	Anexo	75

5.1	Patentes de dispositivos similares.....	75
5.1.1	Sistema y método para controlar automáticamente el movimiento de información en la pantalla.....	75
5.1.2	<i>Mouse</i> Epoc.....	76
5.1.3	Seguidor de la dirección de movimiento de los ojos.....	76
5.1.4	Método y aparato para controlar el movimiento del cursor.....	77
5.2	Programa del dispositivo en el ATMEGA128.....	78
5.3	Notas de aplicación de acelerómetros.....	84
5.3.1	Aceleración rotacional.....	84
5.3.2	Medición de vibraciones	84
5.3.3	Mediciones de golpes.....	86
5.3.4	Descripción de la placa TRIAX	86
5.4	Código de la aplicación de prueba	87
5.5	Descripción del microcontrolador ATMEGA128	89
6.	Referencias y bibliografía.....	95

Índice de ilustraciones

Ilustración 1 Distribución de discapacidad de acuerdo a edad en México	12
Ilustración 2 Porcentaje de acuerdo a discapacidad en México	13
Ilustración 3 Interior de un mouse	15
Ilustración 4 Disco y LEDs infrarrojos.....	16
Ilustración 5 Microcontrolador de un mouse	16
Ilustración 6 Ruedas dentadas.....	17
Ilustración 7 Rueda dentada.....	17
Ilustración 8 Secuencia de estados.....	18
Ilustración 9 Diagrama de fases.....	18
Ilustración 10 Sistema interno del <i>mouse</i>.....	19
Ilustración 11 Emisión, recepción y bloqueo del haz de luz infrarrojo.....	20
Ilustración 12 Desfasamiento de la señal generado al girar al lado izquierdo	20
Ilustración 13 Desfasamiento generado al girar al lado derecho.....	20
Ilustración 14 Diagrama de estados del <i>mouse</i>.....	21
Ilustración 15 Estados del <i>mouse</i>.....	22
Ilustración 16 Estructura de las capas del bus USB	25
Ilustración 17 Capas del sistema de comunicaciones USB	26
Ilustración 18 Vista superior de un control de la consola Wii.....	33
Ilustración 19 Vista Superior del Microcontrolador ATMEGA128	35
Ilustración 20 Disposición de pines del microcontrolador ATMEGA128	36
Ilustración 21 Manera de colocar un acelerómetro de eje X para medir la inclinación.....	39
Ilustración 22 Diagrama de bloques para usar el circuito integrado como inclinómetro.....	39
Ilustración 23 Diagrama de conexiones y de pines para el MMA2260.....	41
Ilustración 24 Modelo físico del transductor y circuito equivalente.....	43
Ilustración 25 Conexión del microcontrolador ATMEGA128	45
Ilustración 26 Diagrama de flujo del programa completo para el ADC con los acelerómetros y los pulsos del mouse.....	46
Ilustración 27 Conexión del driver del mouse	47
Ilustración 28 Diagrama eléctrico del sistema completo (Sin explicación sobre módulos)	49
Ilustración 29 Diagrama eléctrico del sistema completo	49
Ilustración 30 Panel de botones.....	50
Ilustración 31 Vista lateral de la cubierta superior	51
Ilustración 32 Dimensiones de la cubierta superior (vista frontal).....	51
Ilustración 33 Plataforma para el pie derecho	52
Ilustración 34 Salida de los fototransistores al mover el <i>mouse</i>.....	56

Ilustración 35 Imagen obtenida a la salida de un movimiento hacia la derecha	57
Ilustración 36 Imagen obtenida a la salida de un movimiento hacia la arriba	57
Ilustración 37 Imagen obtenida a la salida de un movimiento en diagonal ...	58
Ilustración 38 Función de salida para el acelerómetro 1.	60
Ilustración 39 Intervalo funcional de la salida del primer acelerómetro.....	61
Ilustración 40 Intervalo funcional de la salida del primer acelerómetro.....	63
Ilustración 41 Circuito interno del CAD.....	66
Ilustración 42 Conversión analógico digital	66
Ilustración 43 Rango de operación del CAD en función de la entrada de los acelerómetros.....	67
Ilustración 44 Secuencia de la aplicación de prueba.....	70
Ilustración 45 Sistema y método para controlar automáticamente el movimiento de información en la pantalla.....	75
Ilustración 46 Mouse Epoc	76
Ilustración 47 Seguidor de la dirección de movimiento de los ojos.....	77
Ilustración 48 Método y aparato para controlar el movimiento del cursor	78
Ilustración 49 Programa que determina el desbalance de cargas en una lavadora.....	84
Ilustración 50 Interfaz de monitoreo de vibración.....	85

Índice de Tablas

Tabla 1 Principales causas de la discapacidad en América Latina.	11
Tabla 2 Prevalencia de la discapacidad por país	11
Tabla 3 Paquete de datos enviado por el <i>mouse</i>.....	22
Tabla 4 Pines del conector USB.....	26
Tabla 5 Disposición de pines del circuito MM2260D.....	42
Tabla 6 Costos del prototipo experimental	53
Tabla 7 Cambio en costos de un prototipo precomercial.....	54
Tabla 8 Voltajes de salida (giro hacia la izquierda y hacia la derecha del segundo acelerómetro).....	59
Tabla 9 Valores de salida funcionales correspondientes al primer acelerómetro.....	61
Tabla 10 Voltajes de salida correspondientes a un giro hacia la izquierda y derecha del segundo acelerómetro	62
Tabla 11 Valores de salida funcionales correspondientes al segundo acelerómetro.....	63
Tabla 12 Conversión analógico digital	65
Tabla 13 Pruebas de longitud del cable entre botones y <i>driver</i> USB	68
Tabla 14 Pruebas sensibilidad al movimiento del sistema (horizontal)	69
Tabla 15 Pruebas sensibilidad al movimiento del sistema (vertical).....	69

1. Introducción

1.1 Justificación

En la actualidad, el uso de las computadoras personales se ha incrementado para fines recreativos y laborales. A pesar de que un gran número de personas no hacen uso de esta herramienta, para aquellos que sí lo hacen la pérdida total o parcial de la capacidad motriz en las extremidades superiores representa una disminución en la calidad de vida. Un sistema que permita continuar el uso de una computadora de manera discreta y precisa es algo de gran utilidad para dichas personas.

1.2 Problemática

Una discapacidad puede tener varios orígenes: malformaciones congénitas, accidentes, ataques físicos, deterioro por vejez o enfermedades degenerativas, por mencionar algunos.

Entre las patologías que limitan o impiden la movilidad en las extremidades superiores y que requieren terapia se encuentran las siguientes¹:

- Fracturas
- Lesiones de tendones (lesión tendinosa de la mano)²
 - De los tendones extensores
 - De los tendones flexores
- Lesiones por aplastamiento
- Amputación
- Post quirúrgicas en síndrome del túnel carpiano
- Artroplastias
- Tenorrafia o transposición tendinosas

- Exéresis de tumores y reconstrucción de defectos congénitos
- Síndromes de sobreuso y lesiones relacionadas con el trabajo (epicondilitis, tendinitis)
- Deformidades congénitas
- Lesiones neuropatológicas
- Lesiones del plexo braquial
- Miopatía primaria
- Distrofia muscular
- Enfermedad de Dupuytren³

Las discapacidades son de varios tipos y en el caso de este trabajo se presenta una alternativa a la navegación con un *mouse* de ordenador tradicional para aquellas personas que se encuentran en la situación de discapacidad motriz en miembros superiores ya sea por su estado físico o ausencia de ellos.

Aunados a las restricciones que implica una discapacidad, el sujeto bajo esta condición se enfrenta a otros impedimentos, entre ellos:

- Barreras psicosociales
- Un número reducido de desarrollos en el área
- Difícil acceso a alternativas para realizar tareas cotidianas
- Incumplimiento de las normativas referentes a la población con capacidades diferentes.

Es con el ánimo de brindar herramientas de transformación que logren la reintegración de individuos en situación de discapacidad que se plantea este proyecto.

La importancia de esta reintegración puede ser comparada con la del avance tecnológico de las herramientas computacionales tanto laborales como de entretenimiento y es en la medida en la que el uso del ordenador penetra más y

más en la calidad de vida de la sociedad que hace más importante que todo individuo tenga una participación plena en este ámbito.

El fin último del desarrollo presentado es lograr la igualdad reconociendo las diferencias y propiciar la participación de la comunidad en la creación de estas herramientas.

1.3 Objetivos y propuestas de valor

1.3.1 Objetivos generales

- Diseño de un dispositivo de interfaz humana que sustituya al *mouse* estándar eliminando el uso de las extremidades superiores.
- Lograr la compatibilidad con la mayoría de las PCs.
- Obtener un prototipo experimental seguro y confiable.

1.3.2 Propuesta de valor

- No se hace uso de las extremidades superiores en la navegación.
- Dispositivo estándar para compatibilidad con cualquier PC.
- Funciones idénticas a la de un *mouse* convencional.
- Un dispositivo de accesibilidad como este puede darle una nueva oportunidad laboral a una persona que ha sufrido de pérdida de movilidad

1.4 Alcances y limitaciones

1.4.1 Alcances

- Plataforma compatible con la mayoría de las PCs.
- Movimientos de desplazamiento con puntero
- Alimentación compatible con la PC
- Bajo costo del producto terminado

1.4.2 Limitaciones

- *Driver* de un *mouse* prefabricado
- Destreza, movimiento y estado de las articulaciones inferiores
- Consumo de potencia de los componentes
- Fabricación y ensamblaje a pequeña escala

1.5 Antecedentes

Actualmente existen varios dispositivos que permiten hacer uso de un cursor e interactuar con la computadora sin hacer uso de las extremidades superiores. Estos sistemas emplean, en su mayoría, luces infrarrojas y detección del movimiento de los ojos.

A continuación se enlistan algunos de estos dispositivos, la descripción de cada uno de ellos se encuentra en el anexo (ver sección 5.1):

- Sistema y método para controlar automáticamente el movimiento de información en la pantalla.
- Mouse Epoc
- Seguidor de la dirección de movimiento de los ojos.
- Método y aparato para controlar el movimiento del cursor

1.6 Base teórica

1.6.1 Discapacidad

La definición más general de discapacidad describe este término como la cualidad de discapacitado, siendo este último expresado como:

“Dicho de una persona: Que tiene impedida o entorpecida alguna de las actividades cotidianas consideradas normales, por alteración de sus funciones intelectuales o físicas.”⁴

"Cualquier restricción o carencia (resultado de una deficiencia) de la capacidad de realizar una actividad en la misma forma o grado que se considera normal para un ser humano. Se refiere a actividades complejas e integradas que se esperan de las personas o del cuerpo en conjunto, como pueden ser las representadas por tareas, aptitudes y conductas."⁵

Las discapacidades por su afección se pueden clasificar en física, psíquica o sensorial. Hoy en día la discapacidad se caracteriza por factores que van más allá de sólo la salud, por tal motivo, la clasificación de la discapacidad no pretende ser una clasificación de personas, sino una clasificación de las características de salud de personas que se encuentran en situaciones individuales y su impacto ambiental. El objetivo de la clasificación es identificar con claridad los puntos clave para realizar intervenciones que mejoren el nivel de participación de las personas con alguna discapacidad.⁶

Las causas de una discapacidad pueden ser varias (ver Tabla 1), en el caso de América latina se ubican algunas particulares como el aumento de las enfermedades no transmisibles, conflictos armados, accidentes de todo tipo, uso y abuso de drogas y alcohol, la violencia social, envejecimiento de la población, abandono infantil, la desnutrición, la marginación de grupos como los indígenas, el desplazamiento de la población y los desastres naturales.⁷

Tabla 1 Principales causas de la discapacidad en América Latina.

<i>País</i>	<i>Enfermedades adquiridas</i>	<i>Lesiones causadas por accidentes de tránsito</i>	<i>Lesiones causadas por accidentes laborales</i>	<i>Violencia</i>	<i>Pobreza</i>	<i>Problemas al nacer</i>	<i>Edad</i>
Argentina	X	X	X				
Barbados	X	X					
Bolivia	X	X	X				
Brasil		X	X	X			
Chile	X	X					
Colombia	X	X	X	X			
Costa Rica					X		
Ecuador	X					X	
El Salvador		X			X		
Jamaica		X		X			
México	X	X	X				X
Nicaragua		X		X	X		
Paraguay	X				X		
Perú	X	X	X				
Trinidad y Tabago	X	X					

Fuente: Banco Interamericano de Desarrollo. Informe. Chile: BID, 2001.

La persistencia de la situación de discapacidad en América latina se debe a las causas recién mencionadas. La cantidad de personas con alguna discapacidad para algunos países de América latina se ve en la Tabla 2.

Tabla 2 Prevalencia de la discapacidad por país

<i>País</i>	<i>Total de personas discapacitadas</i>	<i>Porcentaje</i>
Argentina	2 217.500	6,80
Bolivia	741.382	9,26
Brasil	10.000.000	6,00
Chile	958.500	6,39
Colombia	4.992.000	12,00
Costa Rica	363.480	9,32
Ecuador	1 636.800	13,20
Guatemala	1.887.000	17,00
Honduras	700.000	14,00
México	10.000.000	10,00
Nicaragua	593.880	12,12
Perú	7.882.560	31,28
Uruguay	495.000	15,00
Venezuela	2.370.000	10,00

Fuente: Banco Interamericano de Desarrollo. Informe. Chile: BID, 2001.

1.6.2 Discapacidad en México

En el XII Censo General de Población y Vivienda 2000 se definió como una persona con discapacidad a "aquella persona que presenta una limitación física o mental de manera permanente o por más de seis meses que le impide desarrollar sus actividades en forma que se considera normal para un ser humano".

A continuación se describen algunos tipos de discapacidad que afectan a la población en México y datos estadísticos:

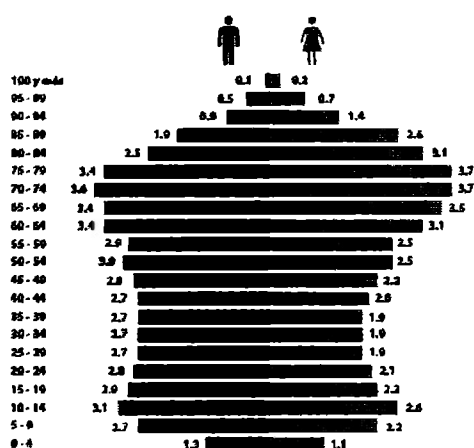


Ilustración 1 Distribución de discapacidad de acuerdo a edad en México
Fuente: <http://www.inegi.gob.mx>

En la Ilustración 1, puede apreciarse que en los grupos de edad de 10 a 14 y de 60 a 79 años se incrementa el porcentaje de personas con discapacidad.

Por el contrario, los puntos más bajos se presentan en los grupos de edad de 0 a 4, de 15 a 39, así como en el de 80 años y más.

1.6.3 Discapacidad Motriz

Se conoce como discapacidad motriz a la pérdida o limitación para moverse, caminar, mantener posturas del cuerpo en su totalidad o de alguna parte.

En el Censo del 2000, se indica que existen ocho personas con discapacidad motriz por cada mil habitantes, esto es: 814 mil personas que representa 45.3% de la población de personas discapacitadas en el país.

Las personas con discapacidad motriz (referirse a la Ilustración 2) se encuentran distribuidas en localidades urbanas y rurales, los que habitan en las urbanas son 76.2% mientras que las que habitan en localidades rurales son un 32.8%.

Por otra parte, cerca de la mitad de la población con discapacidad motriz es mayor de sesenta años. La población económicamente activa representa el 38.5% y la población menor a quince años un 10.1%.⁸

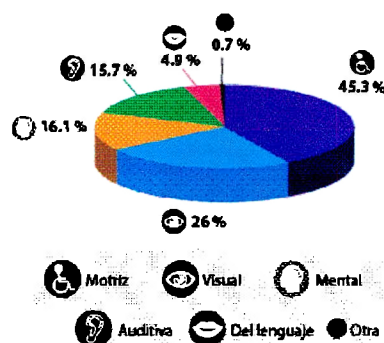


Ilustración 2 Porcentaje de acuerdo a discapacidad en México
Fuente: <http://www.inegi.gob.mx>

1.6.4 Dispositivos de Interfaz Humana

Un dispositivo de interfaz humana, o HID por sus siglas en inglés (*Human Interface Device*), es un tipo de dispositivo para computadoras que interactúa directamente con el ser humano, y que tiene la capacidad de entregar también una salida a los humanos. Esto quiere decir que se tiene una comunicación bidireccional.

Las principales motivaciones para HID eran las de permitir innovaciones en los dispositivos de entrada de la computadora y simplificar el proceso de instalar estos dispositivos. Antes de HID, los dispositivos normalmente se ajustaban a muy

estrictos protocolos para el *mouse* y teclados, entre otros. Cualquier innovación en el *hardware*, requería sobrecargar el uso de datos en un protocolo existente o la creación de un nuevo *driver* y protocolo para los desarrolladores de aplicación. En contraste con todos los dispositivos, los HID entregan paquetes autodescriptivos que pueden contener una infinidad variada de tipos de datos y formatos. Un solo *driver* HID en la computadora traduce los datos y permite una asociación dinámica de datos de entrada y salida (I/O) con la funcionalidad de la aplicación. Este tiene habilitado la innovación rápida y proliferación de nuevos dispositivos de interfaz humana.

El estándar del HID fue desarrollado por una comisión de trabajo con representantes de varias empresas, esta lista de participantes puede ser encontrada en el documento "*Device Class Definition for Human Interface Devices (HID)*"⁹. El concepto de un protocolo autodescribible libre, fue inicialmente concebido por Mike Van Flandern y Manolito Adan, cuando se encontraban trabajando en el proyecto llamado Raptor en Microsoft e independientemente por Steve McGowan mientras trabajaba en el protocolo de un dispositivo para *Access Bus* en Forte. Después de comparar notas en la conferencia de Consumer Game Developer, Steve y Mike aceptaron colaborar en un nuevo estándar para el emergente Universal Serial Bus (USB).¹⁰

Entre los dispositivos HID más comunes se encuentran:

- Teclado
- *Mouse, Trackball, Touchpad, Pointing stick*
- *Joystick*

Mientras que entre los menos comunes destacan:

- Dispositivos de simulación de manejo y vuelo (timón de vuelo, volante y pedales)
- Tapete para baile

1.6.5 Funcionamiento del *mouse*

La meta principal de un *mouse* es traducir el movimiento en señales útiles para la computadora. Un *mouse* realiza la traducción de movimientos en dos ejes del dispositivo en desplazamientos de un cursor en el monitor de una computadora o en algún dispositivo dedicado con capacidad de cómputo y despliegue, por ejemplo, controladores industriales, módulos de Internet con pago por tiempo o videojuegos.

En esta sección se explicará el funcionamiento de un *mouse* optomecánico (ver Ilustración 3), ya que ha sido el empleado como una de las bases para el proyecto desarrollado.

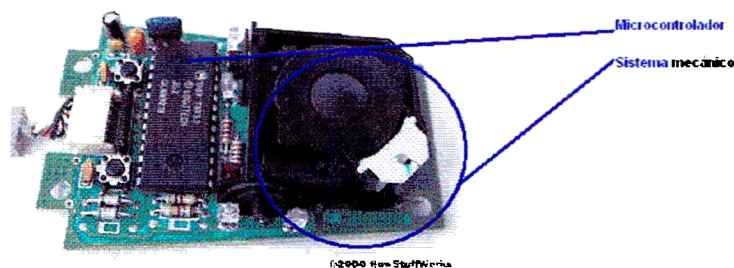


Ilustración 3 Interior de un *mouse*
Fuente: <http://www.howstuffworks.com>

En el *mouse* optomecánico, una pequeña esfera está en contacto con la superficie inferior al dispositivo. Al mover el dispositivo, la esfera rueda y mueve uno o ambos de los cilindros que detectan el movimiento en los ejes X e Y. En uno de los extremos de cada cilindro, hay un pequeño disco con cierto número agujeros que servirán para determinar la dirección del movimiento con ayuda de sensores infrarrojos, como se muestra en la Ilustración 4.

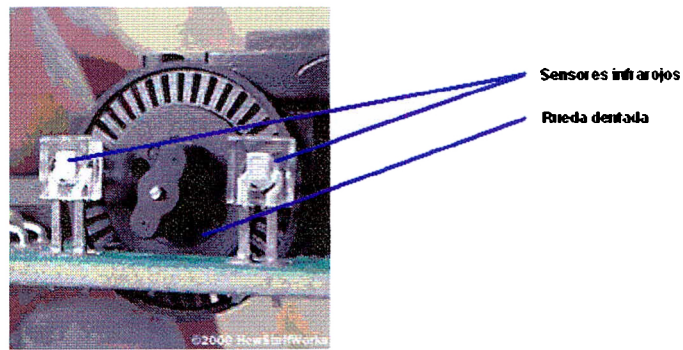


Ilustración 4 Disco y LEDs infrarrojos.
Fuente: <http://www.howstuffworks.com>

La manera en que funciona es moviendo el *mouse* que hace girar un disco, lo que ocasiona que se interrumpa el rayo infrarrojo por lo que el receptor verá las interrupciones como pulsos, que están relacionados directamente con la velocidad en que se mueve el ratón.

Un microcontrolador, colocado sobre la placa de circuito impreso como se observa en la Ilustración 6, detecta estos pulsos y los convierte en datos binarios para que la computadora pueda interpretarlos.

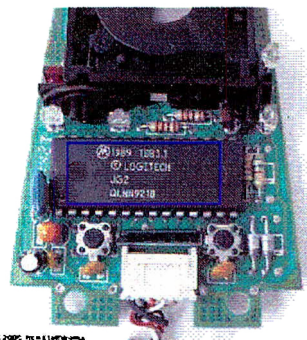


Ilustración 5 Microcontrolador de un mouse
Fuente: <http://howstuffworks.com>

En esta disposición los discos se mueven de manera mecánica y el microcontrolador cuenta los pulsos de la luz. Con el diámetro de la esfera, el del disco y el número de agujeros que tiene el mismo puede determinarse la distancia que recorrió el *mouse*.¹¹

La manera en que se detecta la dirección es por medio de los dos sensores infrarrojos, ya que existe un desfase en los pulsos y así puede detectarse la dirección de movimiento.

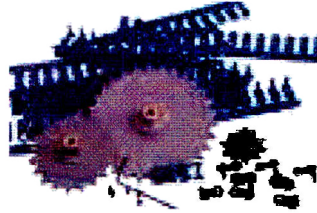


Ilustración 6 Ruedas dentadas
Fuente www.elrafel.com

Dentro de un *mouse* existen dos estructuras similares a las de la Ilustración 6 que son unas ruedas hechas de plástico negro con espacios distanciados uno de otro por cierto número de grados, generalmente la cantidad de grados depende del número de espacios, por ejemplo si son 20 espacios, cada uno está separado del siguiente y del anterior por 18° , para que en total se llegue a 360° . En la Ilustración 4, también existen dos LEDs (representados por los puntos negros) que activan a dos fototransistores correspondientes. Cada LED está separado del otro para que mientras un receptor está encendido, el otro está apagado. En la Ilustración 8, puede apreciarse este efecto. El voltaje de salida del fototransistor es procesado para cambiar rápidamente de alto a bajo, mientras la luz del LED es transmitida u obstruida para que así el voltaje sea bajo mientras el transistor esté prendido y bajo mientras exista "oscuridad". En la Ilustración 7 un LED está iluminando completamente y el otro está cambiando, nótese que el segundo LED puede estar cambiando de luz a oscuridad o de oscuridad a luz, esto depende de la dirección en que está rotando el ratón.

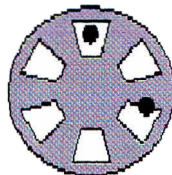


Ilustración 7 Rueda dentada
Fuente: <http://www.4qdttec.com/meece.html>

En la Ilustración 8 se muestran cuatro diferentes estados, cada uno separado cierto número de grados, en este caso 15° , con respecto del anterior. La posición

E es equivalente a la A, únicamente que están rotadas a 60°. Para un movimiento en la dirección de las manecillas del reloj, los estados siguen la siguiente secuencia A → B → C → D → E de izquierda a derecha pero si los estados van de derecha a izquierda siguen la siguiente secuencia: E → D → C → B → A lo que implica un movimiento en dirección contraria a las manecillas del reloj.

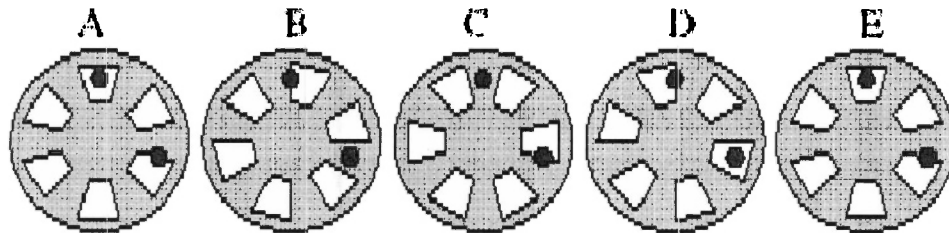


Ilustración 8 Secuencia de estados
Fuente: <http://www.4qdttec.com/meece.html>

Hay que notar que el segundo LED cambia del estado de luz al de oscuridad en A para un movimiento que va en dirección de las manecillas del reloj y en C para un movimiento en contra de las manecillas del reloj. Por lo tanto, si medimos al primer LED cada vez que el segundo LED va de luz a oscuridad, si el primer LED está prendido se está rotando en dirección de las manecillas, pero si el LED está apagado, se está moviendo en la dirección contraria. El *driver* usa este sistema para determinar la dirección, por lo tanto, cada vez que el segundo LED cambia de encendido a apagado. Se monitorea el primer LED para determinar la dirección. El número de cambios también es registrado para determinar la distancia que se movió el ratón. La Ilustración 9 muestra cómo serían las señales habiendo un cambio en intervalos de 15°.

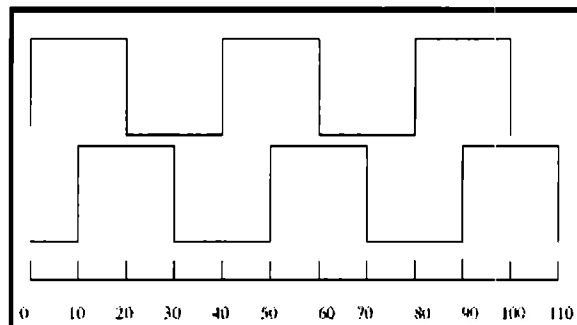


Ilustración 9 Diagrama de fases
Fuente: <http://www.4qdttec.com/meece.html>

En un *mouse* existen los sistemas para detectar el movimiento, uno detecta el movimiento vertical y el otro el horizontal. El ejemplo que se explicó con anterioridad es únicamente para visualizar la manera en que funciona la detección del movimiento. Un *mouse* real tiene muchos más espacios.

A continuación se explicará (ver Ilustración 10) la determinación del movimiento del *mouse*.

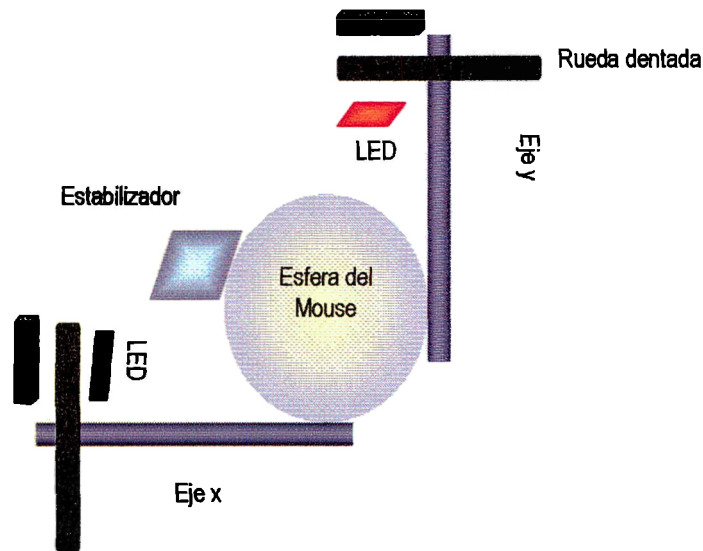


Ilustración 10 Sistema interno del *mouse*

Como se explicó, la esfera hace que se muevan los cilindros, este movimiento causa la interrupción de la recepción de luz por parte del receptor infrarrojo, lo que genera que se mande una señal en forma de pulsos al microcontrolador del *mouse*.

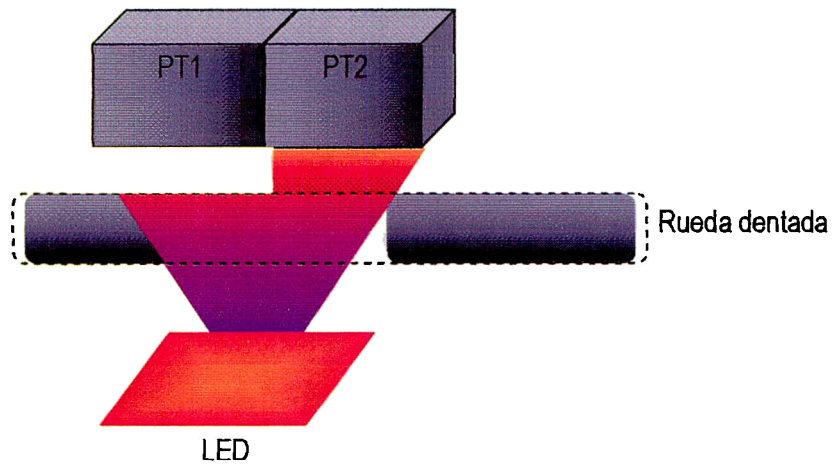


Ilustración 11 Emisión, recepción y bloqueo del haz de luz infrarrojo

Se genera una señal a partir de las interrupciones (Ilustración 11) con la forma de la señal mostrada en la Ilustración 12, indicando el desfase que existe entre un receptor y otro al moverse la rueda dentada hacia la izquierda.

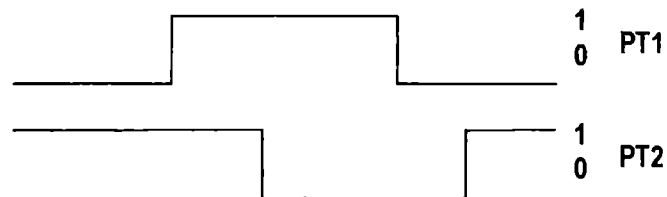


Ilustración 12 Desfase de la señal generado al girar al lado izquierdo

Si el movimiento fuera hacia la derecha se tendría una onda como la siguiente la de la Ilustración 13.

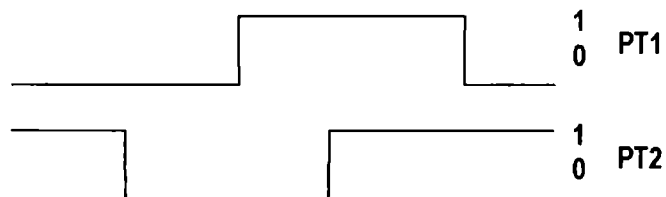


Ilustración 13 Desfase generado al girar al lado derecho

Se identificó el diagrama de estados del funcionamiento del movimiento del *mouse* (Ilustración 14). Consiste en cuatro estados que cambian con respecto a las

entradas (señales de la salida de los transistores). Se nombró a los estados estado 0 (cero), estado 1 (uno), estado 2 (dos) y estado 3 (tres).

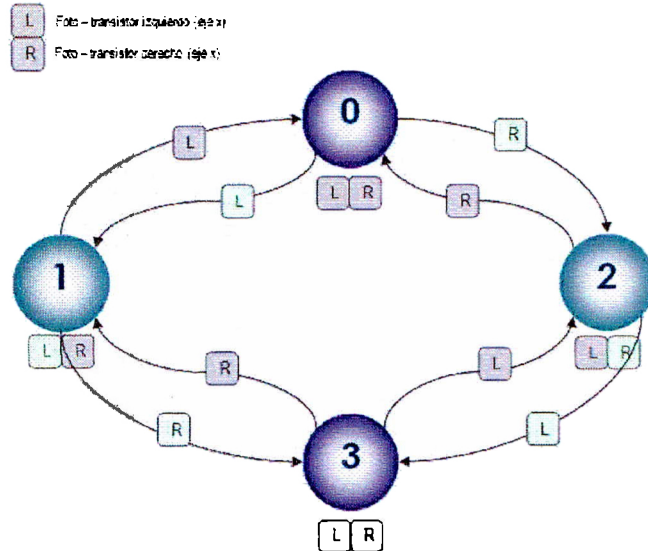


Ilustración 14 Diagrama de estados del *mouse*

La manera en que se identifica el estado siguiente y la dirección del movimiento se describirá a continuación.

Suponiendo que se esté en el estado 0 y se quiera pasar al estado 2, es necesario que en las entradas R (derecha) y L (izquierda) exista un nivel de voltaje alto y bajo respectivamente. Para pasar del estado 2 al 3 es necesario que tanto en R como en L haya un nivel de voltaje alto. Para pasar del estado 3 al estado 1, es necesario que en R haya un estado bajo y en L un estado alto y para ir del estado 1 al estado 0 se necesita un voltaje bajo en R y también en L. Por lo tanto la secuencia sería la siguiente:

$$0 \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow 0$$

Que nos indica un movimiento a la derecha. Y, si se quisiera hacer un movimiento hacia la izquierda, es necesario la secuencia inversa:

$$0 \rightarrow 1 \rightarrow 3 \rightarrow 2 \rightarrow 0$$

Que se obtienen con los niveles de voltaje respectivos para cada estado.

Esto puede apreciarse de mejor manera en la Ilustración 15, donde se muestran las señales L y R con sus estados lógicos¹²

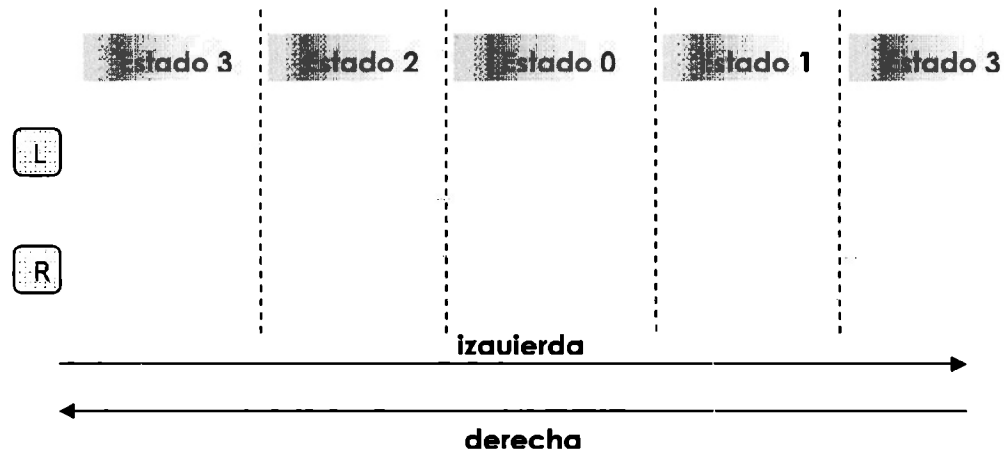


Ilustración 15 Estados del *mouse*

Un *mouse* USB (como el que se usó en este desarrollo) envía los datos de la forma que se muestra en la Tabla 3.

Tabla 3 Paquete de datos enviado por el *mouse*

Bit Byte	D7	D6	D5	D4	D3	D2	D1	D0
1	XV	XV	YS	XS	1	0	R	L
2	X7	X6	X5	X4	X3	X2	X1	X0
3	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0

Y normalmente, aunque puede cambiar entre fabricantes, tiene los siguientes campos:

- L: Botón izquierdo (1 = presionado)
- R: Botón derecho (1 = presionado)
- X0-X7: Movimiento en el eje horizontal
- Y0-Y7: Movimiento en el eje vertical
- XS, YS: Signo del movimiento (1 = negativo)
- XV, YV: Overflow de datos (1 = overflow)

La transmisión bidireccional es controlada por las líneas CLK y DATA. Ambas tienen un dispositivo *open-collector*, que fuerza a que el *mouse* o la PC pongan la línea en "0" lógico. Mientras no se está transmitiendo, la señal CLK está en "1" lógico y la línea DATA puede tener "1" o "0". La computadora puede hacer que el *mouse* no transmita nada únicamente poniendo la línea CLK en "0".

La transmisión de datos tiene las siguientes características:

- Revisar la línea CLK: si está inactiva, entonces hay un bit en la línea DATA.
- Cada transmisión consiste de un bit de inicio (bajo), ocho bits de datos, paridad impar y un bit de paro (alto).
- Cada periodo de actividad o inactividad es de 30 a 50 microsegundos.

Para mandar información se sigue la siguiente secuencia:

- Revisar si las líneas CLK y DATA están en alto,
- poner DATA en bajo para indicar el bit de inicio y
- mandar señal de reloj y datos.¹³

1.6.6 Protocolo USB

El *Universal Serial Bus* (USB) fue creado en 1996 por siete empresas:

- IBM
- Intel
- Northern Telecom
- Compaq
- Microsoft
- Digital Equipment Corporation
- NEC

El estándar incluye la transmisión de energía eléctrica al dispositivo conectado. Algunos dispositivos requieren una potencia mínima, así que se pueden conectar

varios sin necesitar fuentes de alimentación extra. La mayoría de los concentradores incluyen fuentes de alimentación que brindan energía a los dispositivos conectados a ellos, pero algunos dispositivos consumen tanta energía que necesitan su propia fuente de alimentación (muchos discos duros).

El diseño del USB tenía en mente eliminar la necesidad de adquirir tarjetas separadas para poner en los puertos bus ISA o PCI, y mejorar las capacidades *plug-and-play* permitiendo a esos dispositivos ser conectados o desconectados al sistema sin necesidad de reiniciar. Cuando se conecta un nuevo dispositivo, el servidor lo enumera y agrega el *software* necesario para que pueda funcionar.

El USB puede conectar periféricos como *mouse*, teclados, escáneres, cámaras digitales, impresoras, discos duros, tarjetas de sonido, sistemas de adquisición de datos y componentes de red. Para dispositivos multimedia como escáneres y cámaras digitales, el USB se ha convertido en el método estándar de conexión. Para impresoras, el USB ha crecido tanto en popularidad que ha empezado a desplazar a los puertos paralelos porque el USB hace sencillo el poder agregar más de una impresora a una computadora.¹⁴

En el caso de los discos duros, el USB es poco probable que reemplace completamente a los buses como el ATA (IDE) y el SCSI porque el USB tiene un rendimiento un poco más lento que esos otros estándares. El nuevo estándar Serial ATA permite tasas de transferencia de hasta aproximadamente 150/300 MB por segundo. Sin embargo, el USB tiene una importante ventaja en su habilidad de poder instalar y desinstalar dispositivos sin tener que abrir el sistema, lo cual es útil para dispositivos de almacenamiento externos. Hoy día, una gran parte de los fabricantes ofrece dispositivos USB portátiles que ofrecen un rendimiento casi indistinguible en comparación con los ATA (IDE).

El USB no ha reemplazado completamente a los teclados AT y *mouse* PS/2, pero virtualmente todas las placas base de PC traen uno o más puertos USB.

Emplea una topología de estrellas apiladas que permite el funcionamiento simultáneo de 127 dispositivos a la vez (ver Ilustración 16). En la raíz o vértice de las capas, está el controlador anfitrión o *host* que controla todo el tráfico que circula por el bus. Esta topología permite a muchos dispositivos conectarse a un único bus lógico sin que los dispositivos que se encuentran más abajo en la pirámide sufran retardo. A diferencia de otras arquitecturas, USB no es un bus de almacenamiento y envío, de forma que no se produce retardo en el envío de un paquete de datos hacia capas inferiores.

El sistema de bus serie universal USB consta de tres componentes:

- Controlador
- *Hubs* o concentradores
- Periféricos

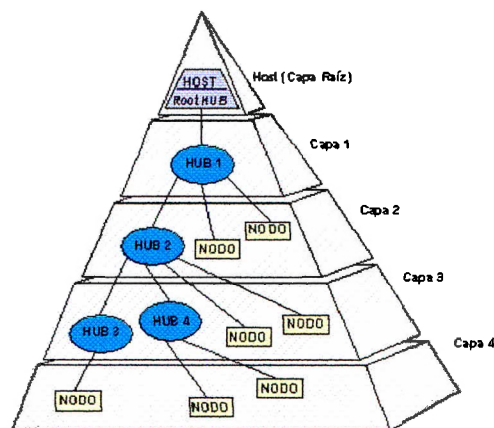


Ilustración 16 Estructura de las capas del bus USB
Fuente <http://www.usb.org>

En el diagrama de capas de la ilustración 17, podemos ver cómo fluye la información entre las diferentes capas a nivel real y a nivel lógico.

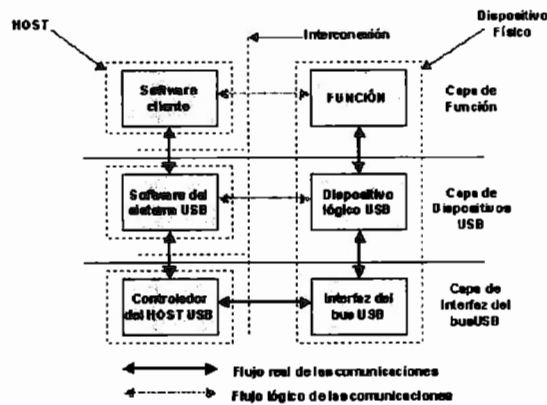


Ilustración 17 Capas del sistema de comunicaciones USB
Fuente <http://www.usb.org>

En la Ilustración 17, también está materializada la conexión entre el controlador anfitrión o host y un dispositivo o periférico. Este está constituido por *hardware* al final de un cable USB y realiza alguna función útil para el usuario.

Las señales del USB son transmitidas en un cable de datos par trenzado con impedancia de $90\Omega \pm 15\%$ llamados D+ y D-. Éstos, colectivamente utilizan señalización diferencial en *half dúplex* para combatir los efectos del ruido electromagnético en enlaces largos. D+ y D- usualmente operan en conjunto y no son conexiones simplex. Los niveles de transmisión de la señal varían de 0 a 0.3V para bajos (ceros) y de 2.8 a 3.6V para altos (unos) en las versiones 1.0 y 1.1, y en $\pm 400\text{mV}$ en Alta Velocidad (2.0). En las primeras versiones, los alambres de los cables no están conectados a masa, pero en el modo de alta velocidad se tiene una terminación de $45\ \Omega$ a tierra o un diferencial de $90\ \Omega$ para acoplar la impedancia del cable.¹⁵

La asignación de los pines del cable USB con terminación tipo A se muestra en la Tabla 4.

Pin	Nombre	Color del Cable	Descripción
1	VCC	Rojo	+5V
2	D-	Blanco	Data-
3	D+	Verde	Data+
4	GND	Negro	Tierra

Fuente <http://www.usb.org>

1.6.7 USB 2.0

Nuevos estándares comenzaron a aparecer y USB 1.1 comenzó a ser desplazado, pues no estaba acorde a las velocidades de transferencia del momento. Así, el puerto IEEE 1394, conocido en el ambiente Mac como FireWire y en los PC como iLink, sobrepasó en velocidad al USB con una tasa de transferencia de 400 Mbps.

En general se pueden enumerar las ventajas que tiene el protocolo USB:

- Conexión más sencilla
- Un sólo tipo de cable
- Gracias al USB prácticamente no se registrarán errores al momento de instalar la impresora, cámara digital o scanner, etc.
- Sólo existe un tipo de cable (EUA: A y B) con conectores distintos en cada extremo, de manera que es imposible conectarlo erróneamente
- *Plug and Play*
- *Hot Pluggable*
- Mayor Rendimiento
- Velocidad
- Mucho más rápido que un puerto serial, casi 3 veces más rápido
- Más rápido que un puerto paralelo
- Mayor capacidad de expansión
- Soporte Multiplataforma
- Responde a todas las necesidades de los usuarios con el mismo *hardware* para todos los dispositivos
- Múltiples dispositivos conectados de manera simultánea
- Es posible conectar hasta 127 dispositivos a nuestra computadora¹⁶

1.6.8 Microsistemas Electromecánicos (MEMS)

Microsistemas Electromecánicos¹⁷, por sus siglas en inglés, se refiere a la tecnología electromecánica micrométrica y sus productos, así como a escalas relativamente más pequeñas (nanométrica) las cuales se fusionan en sistemas nanoelectromecánicos (*Nanoelectromechanical Systems*, NEMS) y Nanotecnología. Los MEMS son independientes y distintos de la hipotética visión de la nanotecnología molecular o Electrónica Molecular. Los MEMS en general varían en tamaño desde un micrómetro a un milímetro. En este nivel de escala de tamaño, las construcciones de la física clásica no son siempre ciertas.

Debido a la gran superficie en relación al volumen de los MEMS, los efectos de superficie como electrostática y viscosidad dominan los efectos de volumen tales como la inercia o masa térmica. El análisis de elementos finitos es una parte importante del diseño de MEMS. La tecnología de sensores ha hecho progresos significativos debido a los MEMS. La complejidad y el rendimiento avanzado de los sensores MEMS ha ido evolucionando con las diferentes generaciones de sensores MEMS.

El potencial de las máquinas muy pequeñas fue apreciado mucho antes de que existiera la tecnología que pudiera construir las. Los MEMS se convirtieron en prácticos una vez que pudieran ser fabricados utilizando modificación de tecnologías de fabricación de semiconductores, normalmente utilizadas en electrónica. Estos incluyen moldeo y galvanoplastia, grabado húmedo y grabado en seco, el mecanizado por electro descarga, y otras tecnologías capaces de fabricar dispositivos muy pequeños.

Existen diferentes tamaños de empresas con importantes programas MEMS. Las empresas más grandes se especializan en la fabricación de componentes de bajo costo y alto volumen o paquetes de soluciones para los mercados finales como el automotriz (detectores de choques para iniciar el inflado de las bolsas de aire),

biomedicina (dispositivos de invasión mínima o nula), electrónica, computación (discos duros, impresoras), dispositivos móviles (celulares, PDAs), entre otros.

El éxito de las pequeñas empresas es ofrecer valor en soluciones innovadoras y absorber el costo de fabricación con altos márgenes de ventas. Tanto las grandes como las pequeñas empresas realizan trabajos de investigación y desarrollo para explorar la tecnología MEMS.

Los avances en el campo de los semiconductores están dando lugar a circuitos integrados con características tridimensionales e incluso con piezas móviles. Los MEMS pueden resolver muchos problemas que un microprocesador más el *software* o configuración no ASIC (chips integrados de aplicación específica) no pueden.

La tecnología MEMS puede aplicarse utilizando un sin número de diferentes materiales y técnicas de fabricación; la elección dependerá del tipo de dispositivo que se está creando y el sector comercial en el que tiene que operar.

Los MEMS son muy utilizados en los campos de los sensores de presión y aceleración, ya que proveen una interfase que es capaz de procesar y/o controlar el entorno. Los MEMS integran dispositivos mecánicos y eléctricos muy diminutos en un único circuito integrado. Son muy utilizados en la electrónica automotriz, equipo médico

Los beneficios que proveen los MEMS son muy variados, van desde el bajo consumo de potencia, pasando por una baja relación costo eficiencia, alto desempeño hasta llegar a un nivel de integración muy pequeño. Debido a la miniaturización, es posible colocar varios dispositivos en el mismo encapsulado, lo que conduce a la disminución de los componentes, así como a un ahorro de costos general.

El sector automotriz tiene el primer puesto en aplicaciones comerciales de los MEMS, ya que se utilizan en muchos aspectos de este sector, por ejemplo:

- Detección de choques
- Medir la presión de las llantas
- Medir protección en caso de volcadura
- Sistemas de seguridad
- Sistemas de frenado
- Entre otros

De manera similar a la industria automotriz, en el área de las tecnologías de información y, en particular, en el área de periféricos para computadora, los MEMS son de uso muy cotidiano, ya que se encuentran en una gran cantidad de dispositivos, por ejemplo:

- Cabeza de impresoras
- Discos duros
- Cartuchos de tinta
- Tarjetas de adquisición de datos
- Entre otros

A pesar de que existan muchas aplicaciones en la industria de la computación, los más grandes prospectos están en el área de las telecomunicaciones, específicamente en interruptores de radio frecuencia y componentes para los dispositivos móviles de nueva generación y aún más importante, en interruptores ópticos y componentes para las redes de transmisión de datos basadas en fibra óptica. Se espera que este sector se convierta en una parte muy importante de consumo de MEMS. Las tecnologías de información se refieren, a groso modo, a la electrónica de transmisión y al procesamiento de información, ya sea numérica,

textual, audio o video y está usualmente asociada a los campos del procesamiento de información, por ejemplo *hardware* y *software* y al de las comunicaciones, específicamente en las telecomunicaciones y en la emisión de señales ya sean aéreas o por cable. Los MEMS hacen posible que cada día salgan al mercado nuevos productos que contenga más y más características (acceso a Internet, GPS, Wi-Fi, cámara, entre otros) y la vez sean más pequeños.

Como resultado del crecimiento de uso de los MEMS en este sector, se han obtenido muchísimos avances en el área de las tecnologías de la información, se han hecho también avances para dispositivos futuros y se ha establecido la necesidad de convertir a tecnología MEMS todo lo referente a sistemas de comunicaciones.

El sector salud también es un sector en el que se ha tenido mucho éxito con la tecnología *MEMS*, ya que se pueden crear dispositivos de tratamiento y diagnóstico que son no invasivos o mínimamente invasivos. La creación de sensores diminutos para poder monitorear algunas variables fisiológicas de manera más precisa y menos molesta para los pacientes. A pesar de estos avances, ésta no es el área en la que esta tecnología tiene sus mayores alcances.

El ámbito que puede ser revolucionado en su totalidad es en las microcirugías, numerosos tipos de implantes, prótesis y *biochips*. Estos tipos de dispositivos son un aspecto de salud pública, ya que sirven para prevenir y evitar que se generen gastos y males mayores. La inversión en prevención es un aspecto que se debe tener muy en cuenta. Ramas de la medicina que usan MEMS son:

- *Biochips*
 - *ADN chip*
 - *Microchips* para microfluidos
- Biosensores
 - Sensores de gas

- Sensores de glucosa
- Sensores de pH
- Sensores de presión
- Bioactuadores
 - Microbombas
 - Microválvulas
 - Micromotores
 - Microagujas
 - Micropinzas
- Bioimplantes
 - Prótesis de oído
 - Retinas artificiales
 - Microelectrodos

Además de los sistemas ya descritos existen otros que son de gran importancia en la actualidad y se hizo uso de ellos en este proyecto. Estos microsistemas son los conocidos como acelerómetros, que se describirán a continuación.

Para la Real Academia Española un acelerómetro es un "aparato que mide la aceleración". Este dispositivo también permite medir el movimiento y las vibraciones a las que está sometido en su modo de medición dinámico, y la inclinación (con respecto a la gravedad), en su modo estático.

De los antiguos acelerómetros mecánicos, de gran tamaño y difíciles de construir, porque incluían imanes, resortes y bobinas (en algunos modelos), ahora se hace con circuitos integrados, con los elementos sensibles creados sobre los propios microcircuitos.

Hay muchas maneras de construir un acelerómetro, por ejemplo, se puede colocar una masa sobre un material piezoeléctrico que, al ser deformado en mayor o

menor medida por los cambios de peso producidos en la masa por la aceleración, se genere una corriente más o menos importante. También los hay basados en tecnología MEMS, y otros que emplean moléculas de gas y una fuente de calor.

En la práctica, son pequeños circuitos integrados que entregan en una de sus salidas (pueden disponer de una para cada eje espacial) una serie de pulsos cuya frecuencia depende de la aceleración a la que están sometidos. Un microcontrolador puede leer esos trenes de pulsos y determinar el sentido y la magnitud de la aceleración. Su precisión es tan grande que se los utiliza para medir vibraciones o, como se ha venido haciendo en la actualidad, para construir diversos tipos de *gadgets*. Un ejemplo de la aplicación de los acelerómetros en productos de consumidor final es el de los controles para videojuegos construidos por la compañía japonesa Nintendo (Ilustración 18) que incorpora la medición de la aceleración en tres ejes para controlar varias funciones en los juegos de su última consola denominada Wii.

Las aplicaciones de los acelerómetros son muy diversas. Algunos aviones de combate incorporan esta tecnología para mantener informados a los pilotos de la intensidad de las fuerzas G a las que están sometidos durante los virajes y las aceleraciones.

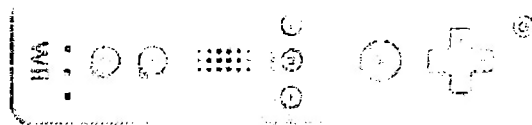


Ilustración 18 Vista superior de un control de la consola Wii.
Fuente: www.cise.ufl.edu

Es gracias al bajo costo de las soluciones que incluyen acelerómetros y a la popularización de pequeños chips que es posible incorporar sistemas de medición de las componentes de aceleración en nuestros propios automóviles y en dispositivos como el *mouse* que se presenta en este trabajo. Información adicional

acerca del funcionamiento y uso de los acelerómetros en conjunto con un sistema de procesamiento se describe en la sección: módulos del sistema: acelerómetros.

1.6.9 Microcontrolador ATMEGA128

Un microcontrolador puede ser una herramienta muy poderosa cuando se van a construir sistemas electromecánicos. Al igual que una computadora, puede ser programado para interactuar con un usuario y con otro *hardware* que esté conectado al microcontrolador. Aún el microcontrolador más sencillo puede realizar las operaciones matemáticas más simples, controlar salidas digitales y monitorear entradas también digitales.

La evolución de estos dispositivos ha ido de la mano con la industria de la computación, ya que los microcontroladores más nuevos son mucho más veloces, tienen mayores capacidades de memoria y un sin fin de nuevas características que minimizan a los microcontroladores anteriores. Algunos de los más modernos tienen integrados otros componentes como convertidores analógico digital, contadores y *timers* de alta velocidad, mayor número de interrupciones, salidas PWM, comunicación con el exterior ya sea vía serial, TWI o SPI entre otras muchas características.

El microcontrolador ATmega128 de Atmel¹⁸, es una de las versiones más recientes de un microcontrolador de 8 bits con arquitectura RISC (set de instrucciones reducidas, por sus siglas en inglés), un microcontrolador como se le consigue comercialmente se puede ver en la ilustración 19. Generalmente se hace uso de un *software* de la misma empresa para poder programar los microcontroladores, en este caso el programa se llama AVRStudio con WinAVR, que es un programa capaz de simular, depurar en tiempo real y programar los microcontroladores de Atmel. Con este programa es factible programar en un lenguaje de alto nivel (C, C++), con lo que se evita el uso de lenguaje

ensamblador, que, a pesar de ser más eficiente, no es capaz de llevar a cabo todas las funciones que un lenguaje de alto nivel puede realizar, además de que el set de instrucciones es específico para el microcontrolador que se esté usando.



Ilustración 19 Vista Superior del Microcontrolador ATMEGA128
Fuente: <http://www.olimex.cl/images/atmega128.png>

El microcontrolador ATmega128 de Atmel cuenta con las siguientes características:

- 128KB de memoria flash interna programable dentro del sistema
- 4KB SRAM interna (expandible hasta 64 KB)
- 4KB EEPROM
- Hasta 16 MIPS (reloj de 16 MHz)
- Arquitectura RISC de 133 instrucciones
- Arquitectura HARVARD (canales de memoria de datos y direcciones separados)
- ALU de 8 bits con los registros cableados directamente
- 32 registros de 8 bits
- Interrupciones externas
- Interfaz SPI
- Interfaz JTAG
- Interfaz TWI
- contadores de 8 bits con preescaladores
- contadores de 16 bits con preescaladores
- 2 canales PWM
- 8 interrupciones externas
- 8 canales multiplexados de ADC de 10 bits, con ganancia

- ± 2 LSB error
- Oscilador RC interno y posibilidad de externo
- Selección del oscilador por medio de software
- 53 líneas I/O en 7 puertos (A-G)
- 2 USARTS
- 6 modos de ahorro de energía

El sistema de interrupciones del ATmega128 tiene como tiempo necesario para ejecutar una interrupción, al menos cuatro ciclos de reloj. Durante estos 4 ciclos el PC es empujado a la pila de datos. Después de estos 4 ciclos, el PC apunta al vector de interrupción correspondiente. Las interrupciones externas se ubican en los puertos PD0:3 (INT0:3) y PE4:7 (INT4:7) como se muestra en la Ilustración 20. Si el sistema de interrupciones se encuentra habilitado, las líneas de entrada ejecutarán las interrupciones aun si los pines son configurados como salidas. El reconocimiento de interrupción por flancos requiere que el reloj CLKI/O se encuentre habilitado.

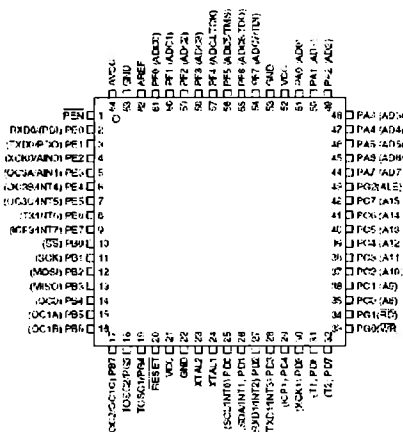


Ilustración 20 Disposición de pines del microcontrolador ATMEGA128
Fuente: http://www.atmel.com/dyn/resources/prod_documents/doc2467.pdf

Para que una interrupción externa sea procesada, se deben cumplir dos condiciones:

- Interruptor Global Definido (Bandera I del SREG)

- Interruptor Individual definido (Registro EIMSK)

-

El control de interrupciones se realiza mediante el uso de 4 registros:

- EIMSK. Interruptor Individual de interrupciones externas
- EICRA. Contiene los bits de las interrupciones 3:0 denominados *Sense Control (SC)*, que definen el tipo de evento que inicializa la interrupción.
- EICRB. Al igual que EICRA, solo que contiene los SC de las interrupciones 4:7.
- EIFR. Cuando un cambio de flanco o un nivel cero lógico, el sistema define en uno el bit correspondiente al número de interrupción que fue llamado. Cuando el procedimiento de interrupción es ejecutado, la bandera es limpiada por hardware.

2. Propuesta de solución

2.1 Módulos del sistema

2.1.1 Acelerómetros

Los acelerómetros de la serie empleada en este proyecto – serie MMA, acelerómetros de silicio capacitivo – tienen varias aplicaciones. Varias de ellas han sido obtenidas de las notas de aplicación de dicha serie.

En seguida se explica la aplicación relacionada con este desarrollo. Otras aplicaciones se encuentran en el anexo (ver 5.3 Notas de aplicación de acelerómetros).

2.1.2 Medición de la inclinación con acelerómetros¹⁹

Una de las características de los acelerómetros es la capacidad que tienen de medir la inclinación de un objeto. Pueden ser usados para medir la aceleración dinámica y estática.

En este caso se hará uso de inclinación estática, ya que la inclinación es una medida que no depende de la aceleración por una fuerza aplicada sino porque la aceleración que se está midiendo es la de la gravedad (9.8 m/s). El sensor MMA2260D tiene una sensibilidad de 1200 mV/g en el eje X (ver Ilustración 21) y la aceleración a la que estará sometido va desde +1g hasta -1g, lo que indica un intervalo de ángulos que van desde -90° hasta +90°.

Para implementar esta aplicación es necesario de un microcontrolador de 8 o 10 bits que tenga 1 o 2 canales de ADC por los cuales se introducirá el voltaje de

salida del acelerómetro. Es recomendable una interfaz gráfica, ya sea en *displays* o una LCD para poder desplegar distintos tipos de información, por ejemplo el ángulo de inclinación. También es posible la implementación de un circuito de comunicación para enviar estos datos a una computadora.

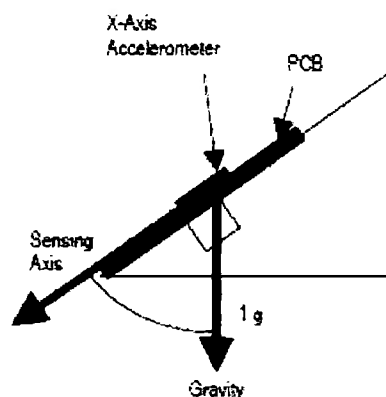


Ilustración 21 Manera de colocar un acelerómetro de eje X para medir la inclinación.
Fuente: <http://freescale.com>

La manera en que se debe colocar el acelerómetro para medir la inclinación se muestra en la Ilustración 22.

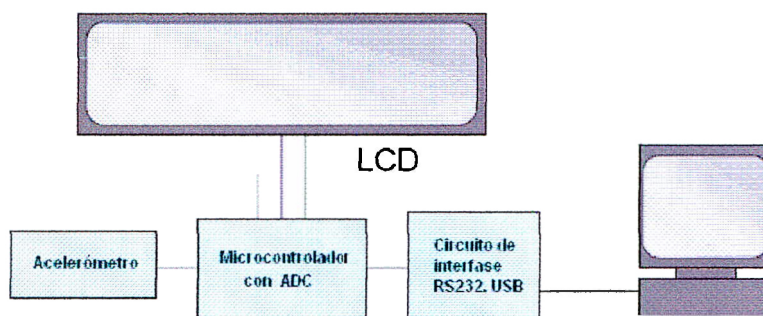


Ilustración 22 Diagrama de bloques para usar el circuito integrado como inclinómetro
Fuente: <http://freescale.com/>

Para obtener la mayor resolución por grado, el eje sensible del circuito debe ser montado de manera paralela al plano en movimiento. Por ejemplo si se quieren medir ángulos desde los 0° hasta los 45°, el circuito debe ser montado perpendicular a la gravedad, por lo que la mejor opción para solucionar este problema será un acelerómetro de eje X.

Existen muchas aplicaciones en las que se requiere medir la inclinación, por ejemplo en el ámbito de los teléfonos celulares la inclinación puede ser usada para controlar las opciones del menú, una brújula electrónica, rotar imágenes o simplemente seleccionar opciones dependiendo de la inclinación. En el ámbito médico, la inclinación sirve para la creación de monitores de presión sanguínea más precisos. También es posible usarlos para determinar la inclinación de las camas o sillas, por lo que un control por medio de inclinación puede ser usado para controlar este tipo de equipo de manera más sencilla. Pueden ser usados también para la rehabilitación y terapia física, en donde los acelerómetros son usados para asegurar el buen uso de los dispositivos, por ejemplo, en un monitor de frecuencia cardiaca para poder determinar que la altura entre el dispositivo y el corazón sean las mismas. Puede generarse un sistema que mida el ángulo del brazo y así guiar al usuario para que tenga una buena postura y así se haga una buena medición. La precisión se incrementa de gran manera al indicarle al usuario en dónde colocar exactamente el brazo. El sistema también puede ser capaz de detener la medición si la posición del brazo cambia. Los acelerómetros también pueden ser usados en controles de videojuegos, dispositivos de entrada para realidad virtual, discos duros portátiles, *mouses* para computadoras y otros dispositivos apuntadores, cámaras, proyectores, lavadoras y sistemas de navegación personal, por mencionar sólo algunos.

Para la transducción de una inclinación en uno de los pies se uso un acelerómetro en modo inclinómetro. El rango de la inclinación va de los -90 a los 90 grados, es decir, noventa grados hacia la izquierda y noventa grados a la derecha.

El circuito construido (ver la Ilustración 23) consiste en un acelerómetro manufacturado por *Freescale* de efecto capacitivo y de un solo eje (modelo MMA2260).

La serie de acelerómetros MMA²⁰ consiste en un conjunto de acelerómetros de silicio capacitivo. Las aplicaciones más comunes en esta serie de acelerómetros

es en el monitoreo de vuelco, inclinómetros, control de dispositivos, monitoreo de orientación mecánica, monitoreo y registro de vibraciones, sistemas y dispositivos de diagnóstico deportivo, sistemas de frenos y aplicaciones automotrices para el consumidor final.

Entre las características principales provistas por este acelerómetro se encuentran:

- Compensación térmica
- Acondicionamiento de señal
- Alta sensibilidad
- Linealidad en la salida
- *Self – test*
- Verificación de paridad de la EPROM
- Sellado hermético
- Robustez en el diseño

El acelerómetro MMA2260 se puede adquirir en diferentes encapsulados. En este proyecto se hizo uso del MMA2260D. Este acelerómetro es de montaje superficial, por lo que fue necesario elaborar un circuito impreso desde el momento de las pruebas.

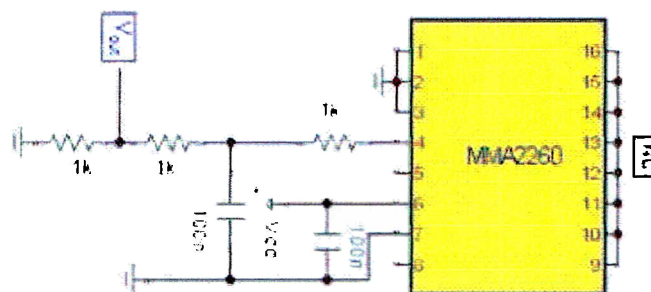


Ilustración 23 Diagrama de conexiones y de pines para el MMA2260

De los 16 pines que componen al CI (ver Ilustración 23) se emplean únicamente ocho, el resto son para sujeción. El mapa de pines tiene dos conexiones redundantes que pueden ser dejadas sin conectar. El resto de los pines son

necesarios para la operación del acelerómetro y para la *self – test*. Entre los pines 6 y 7 se coloca un capacitor para eliminar el ruido eléctrico.

Adicionalmente, a la salida del CI se le conecta un circuito que implementa un filtro pasa bajas compuesto por una resistencia y un capacitor, seguido de un divisor de voltaje que fue usado para obtener los niveles apropiados para su interfase con el microprocesador.

La descripción de los pines es la siguiente:

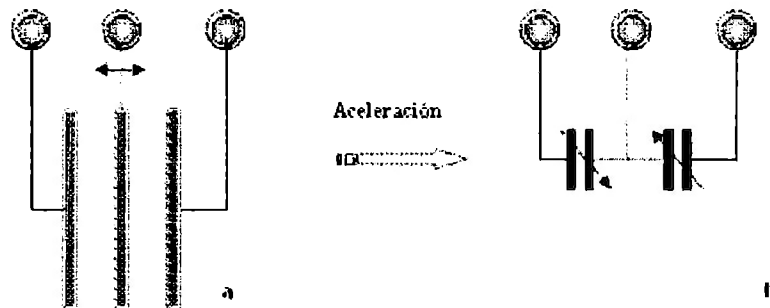
Tabla 5 Disposición de pines del circuito MM2260D

Número de pin	Símbolo	Función
1	VSS	Conexiones internas redundantes a tierra que pueden dejarse sin conectar
2	VSS	
3	VSS	
4	Vout	Voltaje de salida del acelerómetro
5	STATUS	Pin utilizado para indicar fallos
6	VDD	Voltaje de alimentación del sistema
7	VSS	Tierra del sistema
8	ST	Pin para iniciar la auto – prueba
9-16	NC	Sin conexión interna, dejarse sin conectar

Fuente <http://www.freescale.com/>

El dispositivo es sensible a la descarga electrostática, por esta razón, se debe tener precaución durante su manejo. Internamente cuenta con un circuito de protección de descarga electrostática de hasta 2kV. Además de esta consideración se deben tener otras en cuenta como la altura máxima a la que el dispositivo puede dejarse caer, el voltaje de alimentación, la aceleración máxima y la temperatura de almacenamiento.

El dispositivo esta compuesto por una celda de sensado capacitivo²¹, herméticamente sellada, conocida como *g-cell* y un ASIC (circuito integrado de aplicación específica) CMOS que realiza el acondicionamiento de la señal. En la Ilustración 24 se muestra el modelo del transductor utilizado.



a. Modelo físico de la celda g del acelerómetro
 b. Modelo equivalente de la celda g

Ilustración 24 Modelo físico del transductor y circuito equivalente

La *g-cell* es una estructura mecánica formada por materiales semiconductores. Ella puede ser modelada como un conjunto de barras adjuntas a una masa central movable que puede desplazarse entre las barras. Las barras pueden moverse de su posición de reposo, si están sujetas a una aceleración, como puede apreciarse en la Ilustración 24 a. Cuando las barras se mueven, la distancia entre ellas cambia, aumentando de un lado y disminuyendo del otro.

Este cambio en la distancia es una medida de la aceleración. En los acelerómetros de efecto capacitivo, la distancia hace que varíe la capacitancia de la manera siguiente:

$$C = \frac{NA\epsilon}{D}$$

Ecuación 1

En donde N es el número de barras, A es el área frontal, ϵ es la constante dieléctrica y D es la distancia que existe entre las barras. El Circuito Integrado de Aplicación Específica mide la aceleración haciendo una diferencia entre los dos capacitores y también suministra el nivel de voltaje de salida, que es proporcional a la aceleración.

Los acelerómetros MMA contienen un filtro de segundo orden integrado. No es necesario hacer uso de componentes externos para establecer la frecuencia de corte ya que todo se encuentra dentro del circuito.

El sensor tiene una característica que permite verificar la integridad mecánica y eléctrica del acelerómetro en cualquier momento, ya sea antes o después de la instalación. Existe una cuarta barra dentro del la *g-cell* que es utilizada como una placa de prueba. Cuando se le aplica una entrada alta al pin de *self – test*, se aplica un voltaje calibrado a través de la placa de prueba y la placa movable, que crea una fuerza electrostática que hace que en la placa del centro haya una deflexión, que es medida por la ASIC que enviará un nivel de voltaje resultante. Este proceso asegura que todo el acelerómetro está funcionando correctamente.

Los acelerómetros de la familia MMA incluyen una circuitería detectora de fallos así como un *latch* de fallos. El pin de status es una salida que viene del *latch* y una operación lógica de *OR* con *self – test* y está en alto cuando ocurren las siguientes situaciones:

- La paridad de los bits de la EPROM se vuelve impar.
- El *latch* de fallos puede restablecerse por medio de un flanco de subida en el pin de *self – test*, a menos que una o más de las condiciones de fallo existan.

2.1.3 Procesamiento (Microcontrolador)

El montaje del microcontrolador se hizo sobre un circuito impreso que conforma el sistema mínimo que se muestra en la Ilustración 25.

Este cuenta con todas las salidas de puertos, líneas de programación y de alimentación necesarias para el funcionamiento del dispositivo.

Cada uno de los puertos esta formado por ocho pines de entrada salida bidireccionales que normalmente se conectan mediante un cable plano y su conector correspondiente de 8 hilos hacia los dispositivos a controlar o sensor.

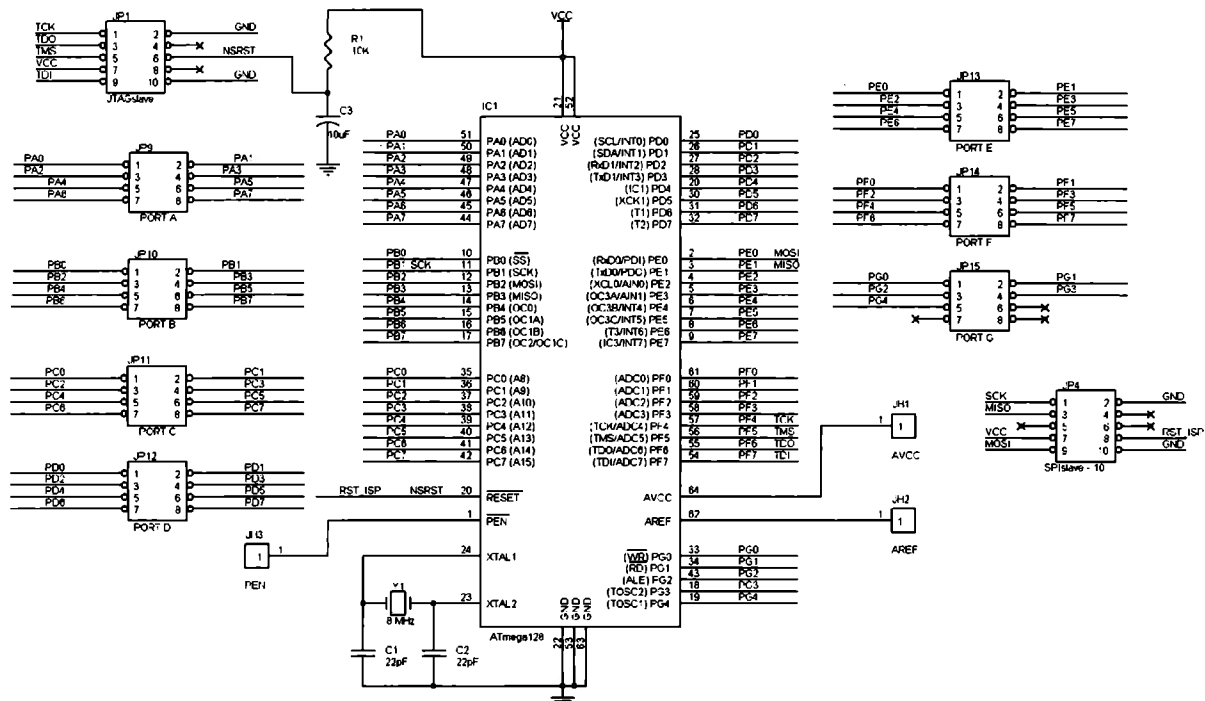


Ilustración 25 Conexión del microcontrolador ATmega128

La placa de circuito impreso sobre la que se coloca el microcontrolador contiene también el oscilador externo del sistema (aunque con frecuencia se emplea el interno a 8 MHz).

En este caso es importante notar que la alimentación debe quedar disponible para la interconexión entre módulos del sistema. De otra manera, los niveles lógicos en la totalidad del sistema serían inconsistentes.

2.1.4 Programación

En la ilustración 26, se muestra el diagrama de flujo del programa en lenguaje C para el sistema completo. El programa al que se refiere el diagrama se encuentra

en el anexo (ver sección 5.2) y hace la conversión analógica digital de las señales del ADC y las convierte a una cantidad en ocho bits con las que trabaja la lógica de decisión del movimiento del cursor en la pantalla. La salida es la emulación de los estados del *mouse* opto - mecánico con el que se ha trabajado durante el proyecto.

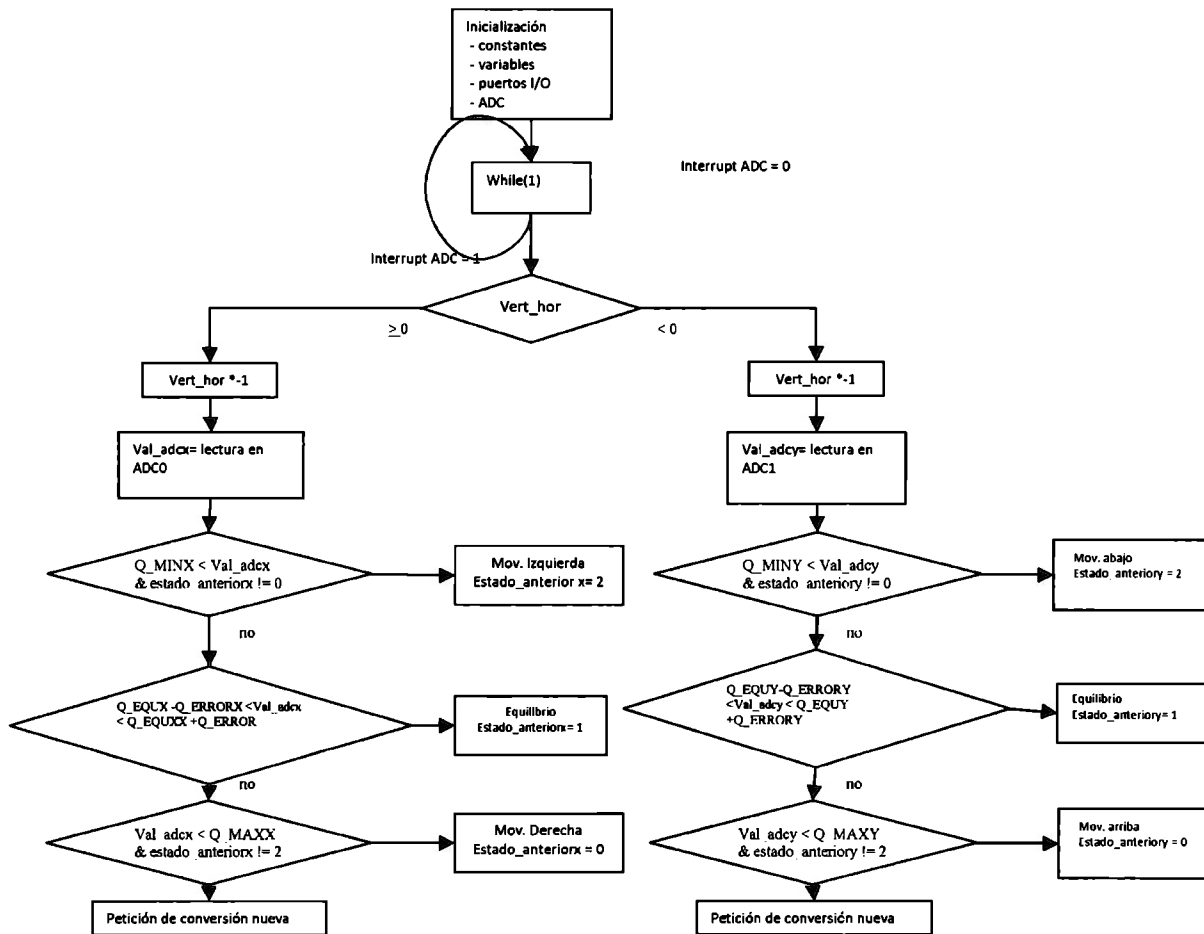


Ilustración 26 Diagrama de flujo del programa completo para el ADC con los acelerómetros y los pulsos del mouse

Un punto clave para el entendimiento del diagrama es conocer el funcionamiento y comportamiento de la señal de salida del acelerómetro empleado con respecto a las entradas posibles. Estas entradas con todos los ángulos posibles de inclinación para el empaque del dispositivo, tanto en el sentido del eje del

acelerómetro como en los otros dos ejes y combinaciones para comprobar que no existen salidas que puedan ser malinterpretadas por el algoritmo.

Otro punto clave es considerar que para la lectura, procesamiento y salida del microcontrolador se efectúa una multiplexión en el tiempo para cada uno de los ejes (el horizontal y vertical) cuyos movimientos son sensados por cada uno de los acelerómetros presentes.

2.1.5 Driver mouse USB

El *driver* utilizado fue el del *MiniMouse* USB que es un circuito integrado de 18 pines en encapsulado DIP. En la Ilustración 27 se muestra el diagrama de conexiones para este módulo.

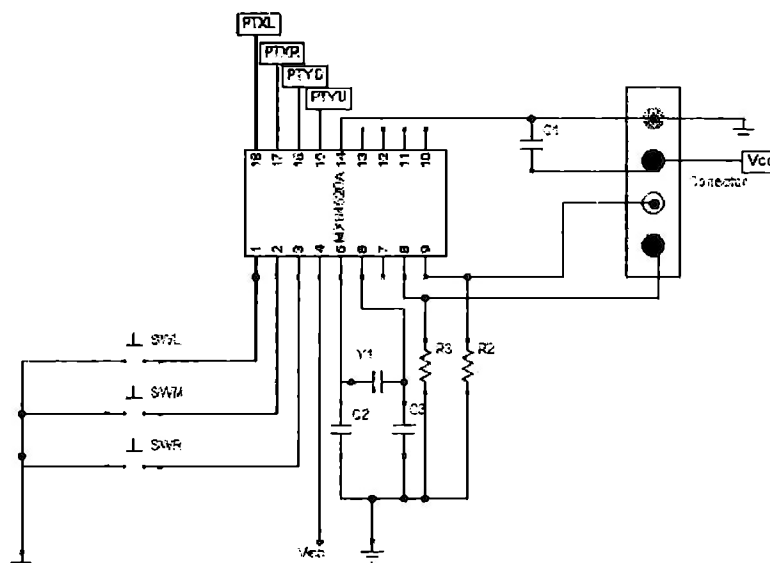


Ilustración 27 Conexión del driver del mouse

Este circuito consiste en tres botones (SWL, SWM y SWR) y cuatro líneas de entrada analógicas que corresponden a las señales enviadas por los foto-

transistores (PTXL y PTXR, para el movimiento horizontal; PTYD y PTYU, para el movimiento vertical).

El CI tiene 4 líneas correspondientes al estándar USB que son: +VCC, GND, Data+ y Data-.

En el caso del *mouse* USB, las líneas de alimentación funcionan como en cualquier otro dispositivo USB y las líneas de dato diferenciales siguen un protocolo serial para el envío de los comandos y detección de movimientos en el plano paralelo a la parte inferior del *mouse*.

2.2 Integración del sistema

2.2.1 Interconexión del sistema

La interconexión es la conjunción de los módulos anteriormente descritos. La observación más importantes probablemente es el seguir un código de colores para los cables y mantener esa consistencia a lo largo del circuito ya que contiene componentes sin tolerancia a la polarización inversa.

Las siguientes figuras corresponden a los diagramas eléctricos del sistema completo.

En la Ilustración 28 se encuentran conectados los dos acelerómetros, uno por cada eje en el plano y-z paralelo al piso, el microcontrolador y el *driver* comercial del *mouse* USB.

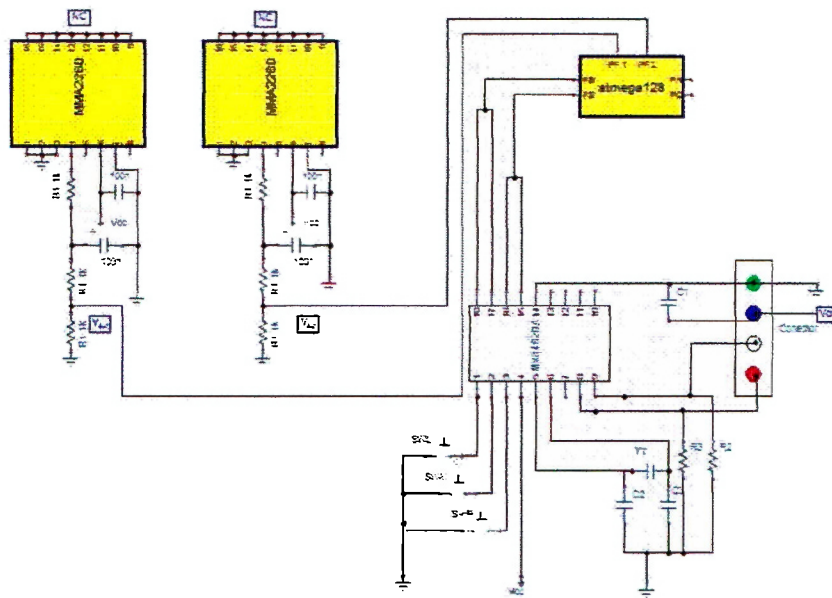


Ilustración 28 Diagrama eléctrico del sistema completo (Sin explicación sobre módulos)

La Ilustración 29, divide a cada uno de los subsistemas y se muestran en un color distinto. Además, se especifica el tipo de señales que se envían y reciben en el módulo que corresponde al microcontrolador.

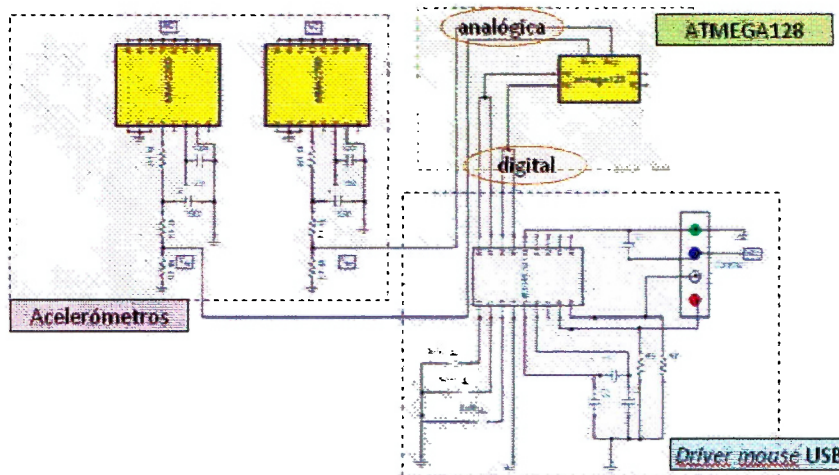


Ilustración 29 Diagrama eléctrico del sistema completo

Entre módulos, la alimentación proviene del mismo par de cables (VCC y GND) provenientes del concentrador USB de la computadora a la que se conecta.

2.2.2 Diseño de la cubierta del prototipo funcional

El prototipo funcional consiste en un panel de botones y una plataforma que debe ser calzada por el pie derecho para poder realizar los movimientos de inclinación que generan el movimiento del *mouse*.

El panel de botones, es manipulado por el pie izquierdo, y a igual que la plataforma semejante a un zapato es colocada en el piso.

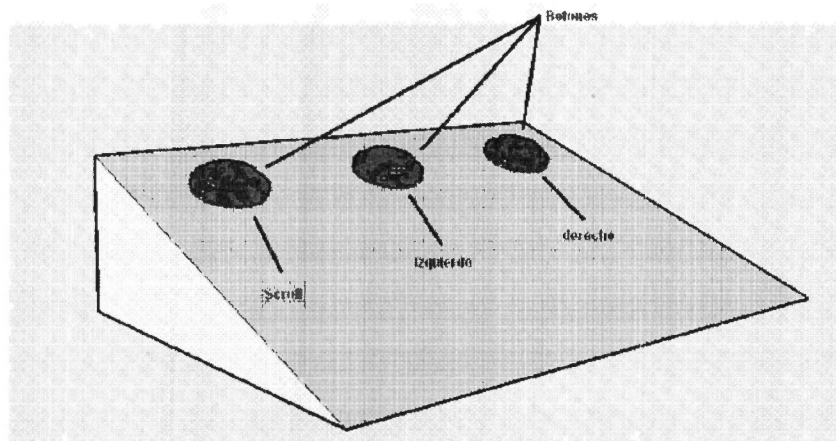


Ilustración 30 Panel de botones

En la Ilustración 30, se muestra el panel de botones completo. Este consiste en tres botones con un radio de 2.6 cm.

La inclinación de la cubierta superior (ver Ilustración 31) con respecto a la superficie sobre la que se coloca el panel es de 20 grados - inclinación recomendada para el apoyo del pie mientras se encuentra una persona sentada -.

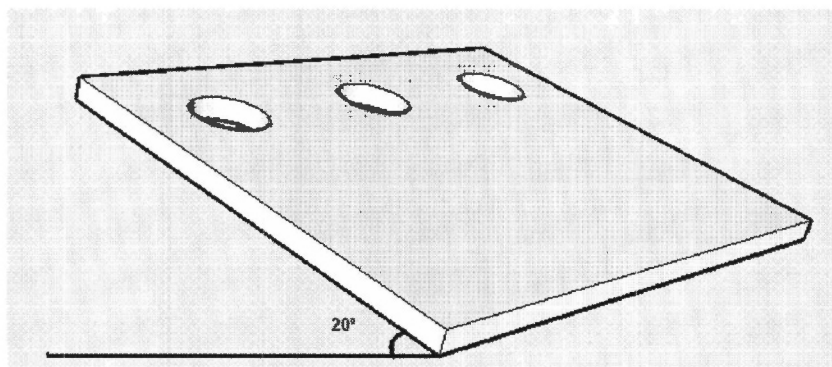


Ilustración 31 Vista lateral de la cubierta superior

Las dimensiones para la cubierta y el radio de los botones, así como la inclinación del pie son esenciales para mantener una buena postura y evitar la fatiga. El primer conjunto de dimensiones se encuentra especificado en el la Ilustración 32.

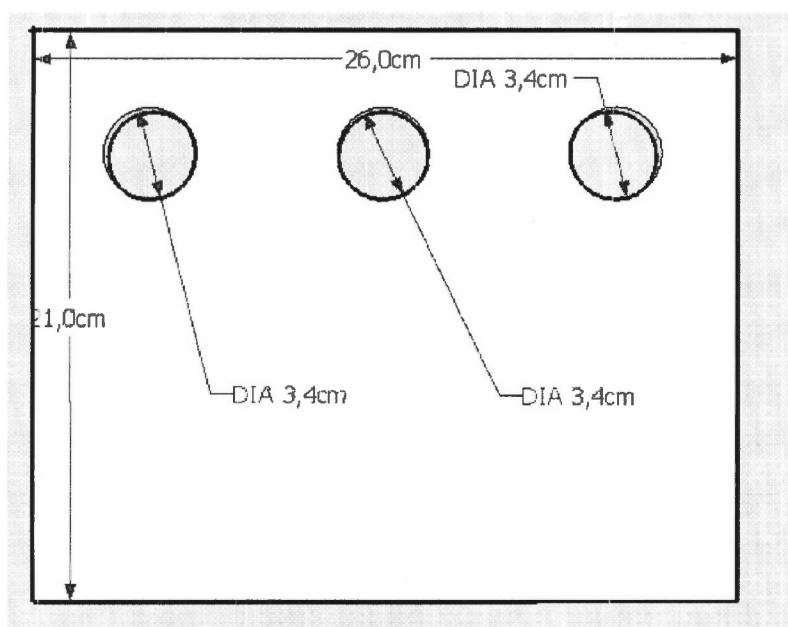


Ilustración 32 Dimensiones de la cubierta superior (vista frontal)

La plataforma a calzar por el pie derecho tiene en la parte inferior un compartimiento para colocar la placa de circuito que contiene a los acelerómetros y el microcontrolador.

En la parte superior es posible insertar el pie y mantenerlo dentro de ella mientras se inclina. Para esto son necesarias las estructuras de sujeción, como se muestra en la ilustración 33.

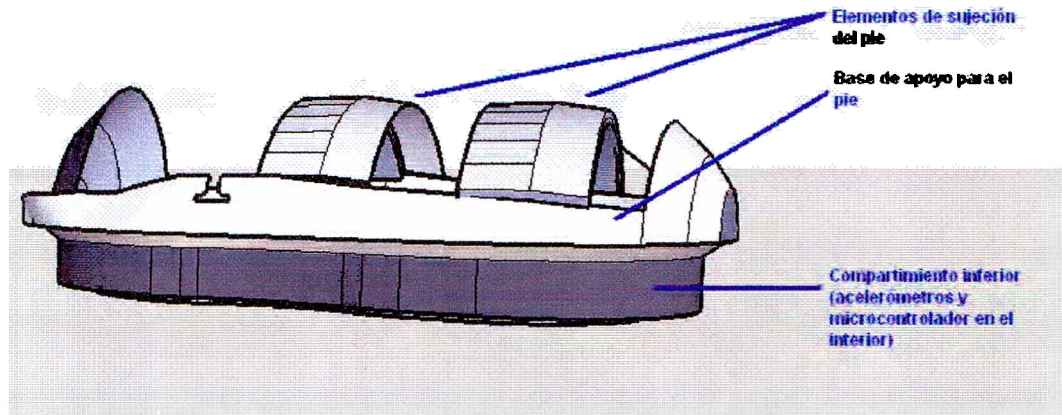


Ilustración 33 Plataforma para el pie derecho

Los materiales usados para el prototipo funcional fueron sólo madera para la totalidad de las cubiertas, excepto para el compartimiento inferior de la plataforma del pie que está construido con PVC y el zapato de EVA (acetato vinilo etileno) ya que son materiales no corrosivos ni tóxicos.

En el diseño de un prototipo comercial o precomercial, la propuesta de materiales para la construcción de las cubiertas sería PVC para las cubiertas del panel de botones y el *driver* USB y una resina de acetato vinilo etileno para la totalidad de la cubierta de la plataforma para el pie. El acetato vinilo etileno, es un material comúnmente utilizado en ingeniería biomédica y como amortiguadores de impacto en zapatos deportivos.²²

2.2.3 Costo del prototipo experimental

En la Tabla 6, se enlistan y describen los componentes utilizados para la construcción del dispositivo completo.

Estos componentes se pueden clasificar en: componentes electrónicos del dispositivo (botones, cable, *driver*, resistencias, capacitores, placas de cobre y acelerómetros) y los de recubrimiento (madera, pintura, bisagras, clavos y el suevo plástico).

Tabla 6 Costos del prototipo experimental

Id.	Componente	Descripción	Costo	Cantidad	Total
1	madera	tríplex	36	1	36
2	pintura	acrílico, acrílico	17.5	2	35
3	bisagras	metálicas 1 pulgada	2	2	4
4	botones	Push button, voltaje $5-12V$ DC	95	15	1425
5	cable	duplex	2	5	10
6	clavos	metálicos 1 pulgada	3.5	8	28
7	driver mouse	de mini mouse USB	60	1	60
8	resistencias	tolerancia de 1%	0.5	8	4
9	capacitores	cerámicos y de políester	3.4	9	30.6
10	placas de cobre	de acelerómetros, 1 microcontrolador ATMEGA, 1 microcontrolador mouse	10	4	40
11	acelerómetros	MMA2260D de freescale	162.5	2	325.2
12	suevo plástico	stick de Eteno Vinyl Acetato (EVA) 4-50 FIE	150	1	150
					2048

El total aproximado para la construcción de un prototipo experimental es de ochocientos cinco pesos mexicanos. Este costo es muy alto para el dispositivo, sin embargo, un prototipo precomercial o comercial implicaría una reducción de costos con la elaboración de moldes y compra de componentes a mayor escala, tal como se muestra en el apartado siguiente.

2.2.4 Costos de un prototipo precomercial

Un prototipo precomercial consistiría en la misma parte funcional de aquel que sirviera como experimental con algunas diferencias.

Una diferencia sería la disposición de los componentes, es posible reducir el número de contenedores de tres a dos lo que reduciría el costo, además de la compra de componentes electrónicos en mayor volumen lo que reduce el costo por componente a aproximadamente la mitad como se muestra en la Tabla 7.

Tabla 7 Cambio en costos de un prototipo precomercial

Precio acelerómetros para más de 1000 unidades (MMA2260EG)	\$5.69 USD
Precio por molde plástico (FVC)	\$25.000 - \$30.000 m.n.
Precio por pieza plástica (a partir de dichos moldes)	\$4,0 - \$5,0 m.n.
Precio del microcontrolador USB para más de 1000 unidades (Freescale MCHC908JW32FC)	\$3.15 USD

Existen algunos otros cambios a nivel de implementación que sería posible realizar para que el dispositivo realizara funciones adicionales como monitoreo de movimientos y un mejor acoplamiento con paquetes de software de uso común.

3. Resultados y Pruebas

3.1 Ingeniería inversa y adaptación del *driver* USB

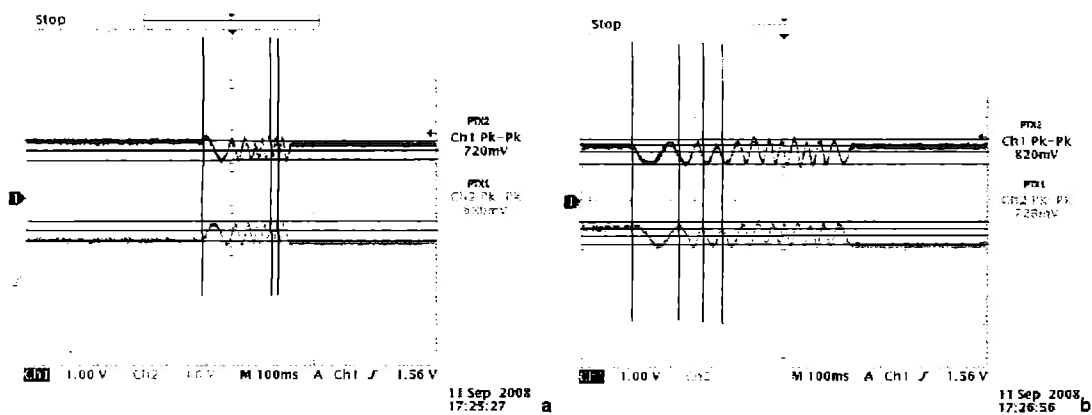
Se investigó el uso de *mouse* para poder determinar el tipo de señales que recibe el controlador para lo que se hicieron varias pruebas. Una de ellas fue la de graficar con un osciloscopio la salida que había en los fototransistores, con el fin de determinar la señal que nosotros debíamos emular. Al graficar las señales de los dos fototransistores, pudimos apreciar que existe un desfase entre las señales.

Este desfase es indispensable para poder determinar la dirección del movimiento del *mouse*. Se removieron los LEDs infrarrojos así como los fototransistores para poder introducir, por medio del microcontrolador en esas líneas del *driver*, la secuencia de estados correspondiente a los distintos movimientos que tiene el *mouse*.

Otra prueba en que los fototransistores estuvieran bloqueados, esto con el fin de determinar el nivel de voltaje que emitía al estar totalmente tapado, simulando así el estar bloqueado por la rueda dentada.

Posteriormente se iluminó totalmente, con lo que se obtuvo el valor del voltaje cuando una apertura de la rueda dentada estaba entre el LED infrarrojo emisor y el fototransistor.

Las imágenes que se obtuvieron durante las pruebas del *driver* se muestran a continuación:



a. Señal analógica del movimiento del *mouse* con una velocidad alta
 b. Señal analógica del movimiento del *mouse* con una velocidad baja
 Ilustración 34 Salida de los fototransistores al mover el *mouse*

Como puede apreciarse en la Ilustración 34 a y b, la frecuencia con la que se emite la señal del transistor es proporcional a la velocidad con la que se mueve el *mouse*, esto se debe a que entre más rápido se mueve el ratón, más rápido girará la esfera que está dentro y por consiguiente la rueda dentada se moverá más rápido, con lo que el fototransistor recibirá el haz luminoso con una frecuencia mayor.

Podemos notar también que el estado sistema mantiene el estado, ya que si el *mouse* está en reposo, la rueda dentada no se mueve y, por ende, el fototransistor está recibiendo siempre la misma cantidad de luz.

Las dos imágenes (Ilustración 34 a y b) nos muestran también el desfaseamiento que existe de una señal con respecto a la otra, lo que nos indica la dirección del movimiento. En las imágenes de arriba puede apreciarse con claridad el cambio de estados. En la segunda imagen se tenía inicialmente el estado 3, posteriormente se movió al estado 1, luego al estado 0, luego al estado 2, con lo que se puede determinar que se trata de un movimiento hacia la derecha.

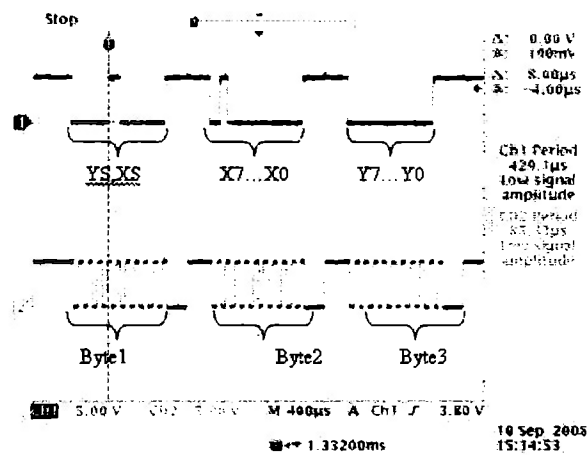


Ilustración 35 Imagen obtenida a la salida de un movimiento hacia la derecha

En la Ilustración 35, puede apreciarse lo que se había comentado con anterioridad acerca del formato de la transmisión de datos. Los bits que representan al movimiento en el eje vertical se encuentran en cero, ya que no existe movimiento vertical, sólo horizontal.

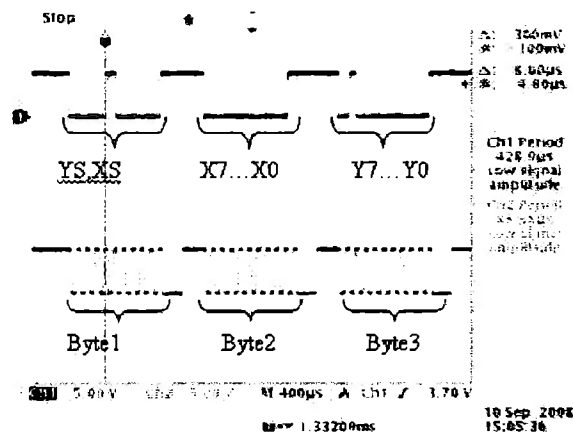


Ilustración 36 Imagen obtenida a la salida de un movimiento hacia la arriba

En la Ilustración 36, puede apreciarse lo que se había comentado con anterioridad acerca del formato de la transmisión de datos. Los bits que representan al movimiento en el eje horizontal se encuentran en cero, ya que no existe movimiento horizontal, sólo vertical.

Puede distinguirse con claridad que la transmisión de datos se hace de acuerdo al formato que habíamos mencionado con anterioridad. Se tienen 3 bytes para determinar el estado del mouse a la computadora. Con las imágenes anteriores, puede deducirse que un movimiento diagonal es la suma de un movimiento horizontal y uno vertical, por lo que se intuye que en los bytes correspondientes a [X7...X0] e [Y7...Y0] tendrán información. Esto pudo constatarse con la siguiente imagen, que corresponde a un movimiento en diagonal (Ilustración 37)

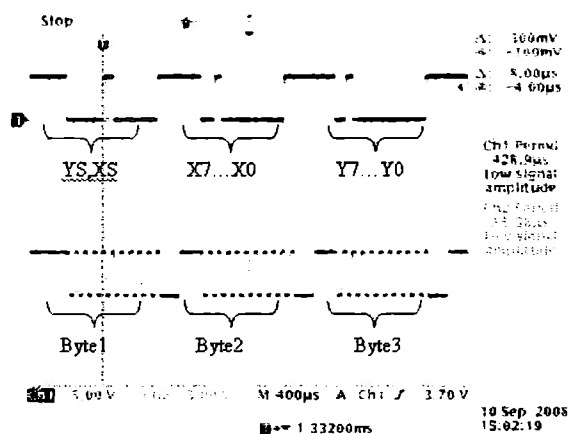


Ilustración 37 Imagen obtenida a la salida de un movimiento en diagonal

3.2 Caracterización del acelerómetro como inclinómetro

Para poder determinar los rangos de operación del acelerómetro MM2260D fue necesario hacer muchas mediciones para obtener los voltajes que da el MEM al estar inclinado cierto número de grados. Las pruebas que se hicieron fueron las siguientes:

- Inclinarse el acelerómetro hacia la izquierda desde 0° hasta 360° (medidos con un transportador a intervalos de 15°).
- Inclinarse el acelerómetro hacia la derecha desde 0° hasta 360° (medidos con un transportador a intervalos de 15°).
- Se realizaron las mismas pruebas en dos acelerómetros para determinar sus curvas de comportamiento.

Los resultados obtenidos se muestran en las Tabla 8. Para su realización se hicieron 50 mediciones, 25 hacia la derecha y 25 hacia la izquierda. Esto para cada uno de los acelerómetros.

Tabla 8 Voltajes de salida (giro hacia la izquierda y hacia la derecha del segundo acelerómetro)

Medición #	Giro hacia la izquierda		Giro hacia la derecha	
	Ángulo [°]	Salida [V]	Ángulo [°]	Salida [V]
1	0	2.34	0	2.34
2	15	2.2	15	2.66
3	30	1.9	30	2.86
4	45	1.54	45	3.16
5	60	1.38	60	3.36
6	75	1.26	75	3.52
7	90	1.2	90	3.54
8	105	1.24	105	3.52
9	120	1.32	120	3.38
10	135	1.48	135	3.24
11	150	1.72	150	3.06
12	165	2.02	165	2.78
13	180	2.3	180	2.42
14	195	2.66	195	2.2
15	210	2.92	210	1.92
16	225	3.12	225	1.56
17	240	3.26	240	1.38
18	255	3.42	255	1.26
19	270	3.46	270	1.24
20	285	3.46	285	1.3
21	300	3.36	300	1.42
22	315	3.18	315	1.56
23	330	3.12	330	1.76
24	345	2.76	345	1.96
25	360	2.34	360	2.36

Al graficarse se obtuvo el resultado de la Ilustración 38.

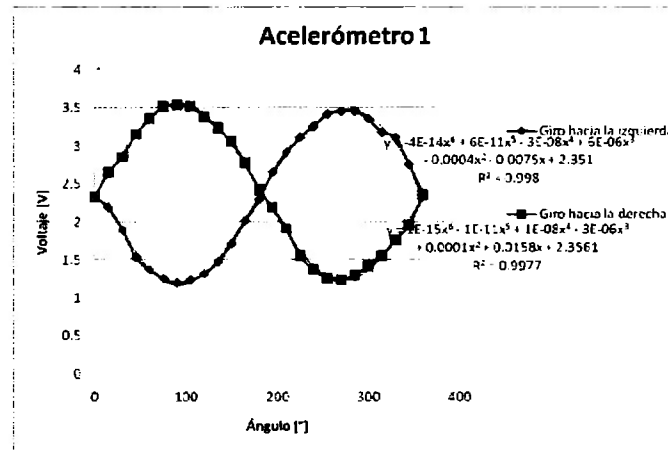


Ilustración 38 Función de salida para el acelerómetro 1.

Como puede apreciarse en la Ilustración 38, el acelerómetro se comporta como una función aproximable a un polinomio de orden 2 en ciertos intervalos.

Debido a que se toman en cuenta valores de referencia para el equilibrio, la izquierda o la derecha (arriba y abajo, análogamente para el segundo acelerómetro), basta con que la función cumpla con ser monotónicamente creciente en el intervalo de -90 a 90 °. Dado que cumple con esta condición, la resolución y la linealidad de la salida no son factores que impidan el funcionamiento correcto del algoritmo programado en el microcontrolador en caso que de que no tuviera estas características.

Se utilizaron únicamente los intervalos que van desde 90 ° hacia la izquierda hasta 90 ° hacia la derecha. Esto debido a que la flexión del pie no va más allá de estos límites.

Tomando en cuenta sólo el intervalo de interés, se tienen los siguientes valores para el primer acelerómetro:

Tabla 9 Valores de salida funcionales correspondientes al primer acelerómetro

Medición #	Giro hacia la izquierda		Giro hacia la derecha	
	Ángulo [°]	Salida [V]	Ángulo [°]	Salida [V]
1	0	2.34	0	2.34
2	15	2.2	15	2.66
3	30	1.9	30	2.86
4	45	1.54	45	3.16
5	60	1.38	60	3.36
6	75	1.26	75	3.52
7	90	1.2	90	3.54

Al graficar los valores de la Tabla 9, puede apreciarse que la zona de utilización es creciente, por lo que predecir el valor que tendrá la salida es predecible y no habrán errores de sobre escritura de datos al convertir el voltaje analógico en un valor digital.

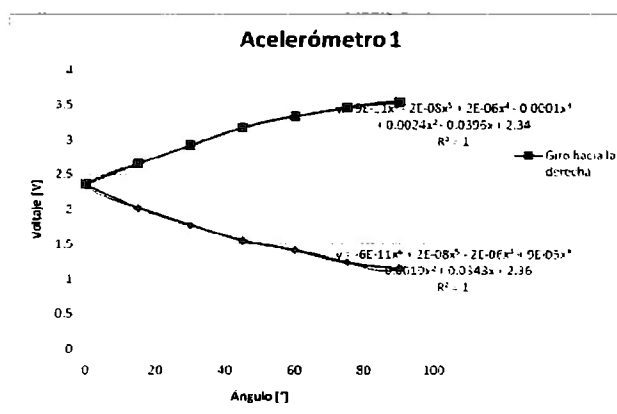


Ilustración 39 Intervalo funcional de la salida del primer acelerómetro

Para la caracterización del otro acelerómetro se siguieron los mismos pasos y se obtuvieron los siguientes resultados:

Tabla 10 Voltajes de salida correspondientes a un giro hacia la izquierda y derecha del segundo acelerómetro

Medición #	Giro hacia la izquierda		Giro hacia la derecha	
	Ángulo [°]	Salida [V]	Ángulo	Salida [V]
1	0	2.34	0	2.36
2	15	2.02	15	2.66
3	30	1.78	30	2.92
4	45	1.56	45	3.18
5	60	1.42	60	3.34
6	75	1.24	75	3.46
7	90	1.16	90	3.54
8	105	1.18	105	3.48
9	120	1.28	120	3.36
10	135	1.46	135	3.14
11	150	1.74	150	2.96
12	165	2.06	165	2.66
13	180	2.3	180	2.34
14	195	2.66	195	2.06
15	210	2.94	210	1.74
16	225	3.2	225	1.48
17	240	3.36	240	1.36
18	255	3.46	255	1.2
19	270	3.52	270	1.16
20	285	3.5	285	1.22
21	300	3.4	300	1.28
22	315	3.18	315	1.5
23	330	2.98	330	1.74
24	345	2.7	345	2.04
25	360	2.34	360	2.36

Se graficaron los valores y se obtuvo la siguiente curva, que al igual que la anterior puede apreciarse que el acelerómetro tiene una salida creciente.

Tomando en cuenta lo mencionado con anterioridad, se tienen los siguientes valores para el segundo acelerómetro:

Tabla 11 Valores de salida funcionales correspondientes al segundo acelerómetro

Medición #	Giro hacia la izquierda		Giro hacia la derecha	
	Ángulo [°]	Salida [V]	Ángulo [°]	Salida [V]
1	0	2.34	0	2.36
2	15	2.02	15	2.66
3	30	1.78	30	2.92
4	45	1.56	45	3.18
5	60	1.42	60	3.34
6	75	1.24	75	3.46
7	90	1.16	90	3.54

Por lo que al graficarse se obtiene la siguiente curva (Ilustración 40)

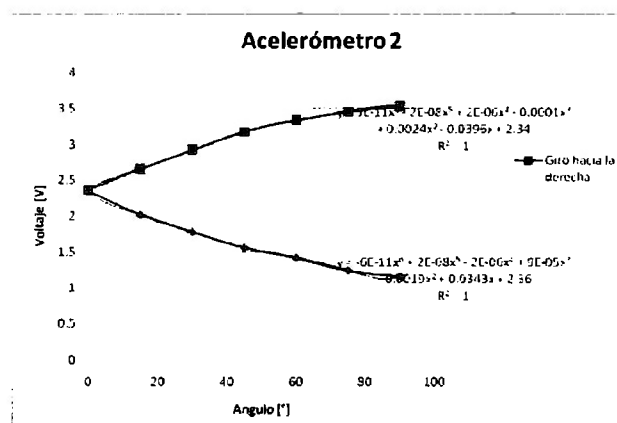


Ilustración 40 Intervalo funcional de la salida del primer acelerómetro

En la gráfica (Ilustración 40) puede apreciarse que ese es un intervalo útil, por lo que predecir el valor que tendrá la salida es perfectamente predecible y, nuevamente, no habrán errores de sobre escritura de datos al convertir el voltaje analógico en un valor digital.

3.3 Caracterización del Convertidor Analógico Digital

Para la caracterización del convertidor analógico digital del microcontrolador empujado, se hicieron las pruebas correspondientes a la conversión de señales analógicas producidas por una fuente variable, la comparación entre los convertidores del microcontrolador, pruebas con el acelerómetro.

La manera en que el ATmega128 hace las conversiones es por el método de aproximaciones sucesivas.

Se hicieron pruebas con dos de los 8 convertidores analógico digital que tiene el ATmega128 en el puerto F y se obtuvieron los resultados mostrados en la Tabla 12.

En la Tabla 12 también puede apreciarse que la conversión analógico digital no varía respecto al número de CAD que se seleccione, esto es debido a la estructura interna del microcontrolador (ver Ilustración 41), ya que la unidad de conversión es la misma, lo único que cambia es la señal de entrada (multiplexaje).

Tabla 12 Conversión analógico digital

Medición	Entrada PF.0 [V]	Valor Binario	Entrada PF.1 [V]	Valor Binario
1	0.005	2	0.007	3
2	0.106	10	0.108	11
3	0.211	22	0.207	20
4	0.298	31	0.301	30
5	0.41	40	0.407	40
6	0.505	47	0.501	48
7	0.605	60	0.61	61
8	0.709	76	0.698	75
9	0.8	84	0.802	84
10	0.895	92	0.9	92
11	0.997	104	0.995	104
12	1.102	110	1.1	109
13	1.199	120	1.194	119
14	1.302	127	1.298	126
15	1.405	143	1.407	143
16	1.5	154	1.503	154
17	1.596	158	1.599	158
18	1.708	172	1.702	171
19	1.802	185	1.798	184
20	1.902	189	1.899	188
21	2.013	193	2.01	193
22	2.1	216	2.102	216
23	2.205	218	2.196	217
24	2.298	220	2.302	221
25	2.398	233	2.403	234
26	2.497	251	2.5	250
27	2.696	255	2.701	255

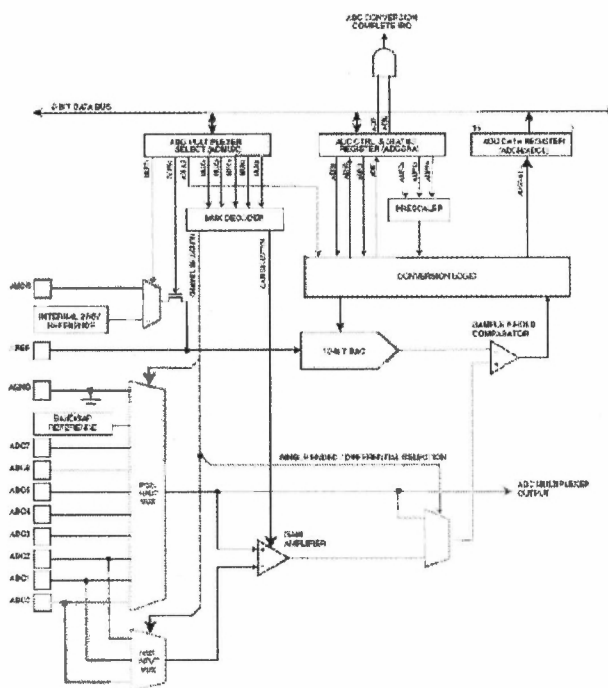


Ilustración 41 Circuito interno del CAD

La gráfica de los datos anteriores se muestra en la Ilustración 42 que contiene una representación de los diferentes valores medidos en las entradas analógicas a un valor digital.

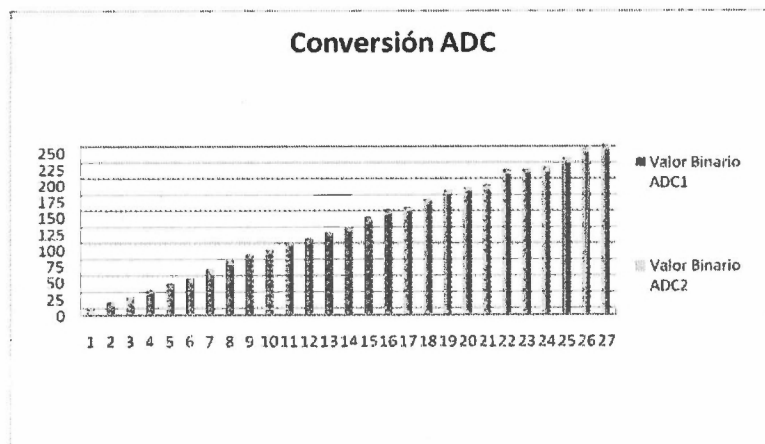


Ilustración 42 Conversión analógico digital

Las variaciones que se observan son debido al voltaje que se aplicaba al puerto, ya que como se mencionó con anterioridad, el módulo del CAD es el mismo y lo único que cambia es la señal de entrada.

La referencia del convertidor analógico digital fue fijada internamente a 2.56V, por lo que fue necesario hacer un divisor de voltaje para reducir el voltaje aplicado a la entrada del puerto. Las resistencias que se usaron para lograr este propósito fueron de 1K Ω , tomando ahora el voltaje de entrada al microcontrolador, la terminal central del divisor.

Con esto se acotó el rango de operación del CAD al intervalo de 0.2V hasta 1.6V, por lo que la gráfica queda con la de la Ilustración 43.

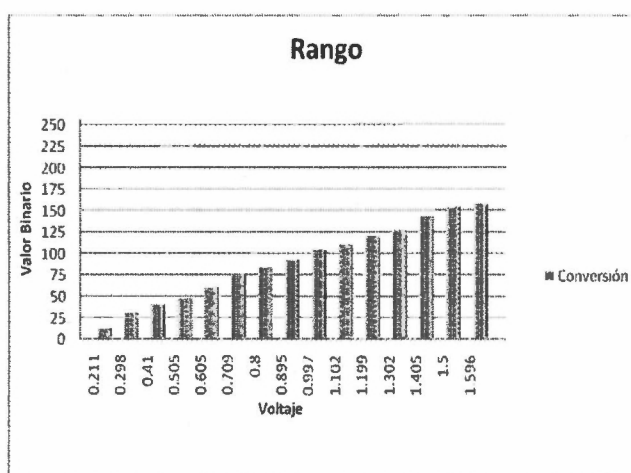


Ilustración 43 Rango de operación del CAD en función de la entrada de los acelerómetros

Una vez que se obtuvieron los valores digitales, se pudo continuar con el programa y así pasar a la integración de todo el sistema.

3.4 Pruebas de longitud del cable

Para la interconexión entre los módulos de botones de control y el *driver* del *mouse* USB se realizaron pruebas para determinar si la longitud del cable (ver Tabla 13) máximo a instalar en el dispositivo funcionaría de manera correcta, es decir, no se perdería el nivel de voltaje por la longitud ni existieran entradas falsas procesadas por el microcontrolador.

Tabla 13 Pruebas de longitud del cable entre botones y *driver* USB

No. de prueba	Botón presionado	Longitud del cable (cm)	Resultado
1	Izquierdo	5	<input checked="" type="checkbox"/>
2	Izquierdo	50	<input checked="" type="checkbox"/>
3	Izquierdo	100	<input checked="" type="checkbox"/>
4	Izquierdo	150	<input checked="" type="checkbox"/>
5	Izquierdo	200	<input checked="" type="checkbox"/>
6	Izquierdo	250	<input checked="" type="checkbox"/>
7	scroll	5	<input checked="" type="checkbox"/>
8	scroll	50	<input checked="" type="checkbox"/>
9	scroll	100	<input checked="" type="checkbox"/>
10	scroll	150	<input checked="" type="checkbox"/>
11	scroll	200	<input checked="" type="checkbox"/>
12	scroll	250	<input checked="" type="checkbox"/>
13	scroll	5	<input checked="" type="checkbox"/>
14	derecho	50	<input checked="" type="checkbox"/>
15	derecho	100	<input checked="" type="checkbox"/>
16	derecho	150	<input checked="" type="checkbox"/>
17	derecho	200	<input checked="" type="checkbox"/>
18	derecho	250	<input checked="" type="checkbox"/>

De estos resultados podemos concluir que es posible conectar el dispositivo a una distancia de 250 centímetros aún con la confiabilidad de que las entradas del dispositivo corresponderán con la salida esperada.

3.5 Prueba de sensibilidad del desplazamiento

En las Tablas 14 y 15, se muestran la tolerancia del ángulo para permanecer en equilibrio y el ángulo mínimo al que se debe inclinar el dispositivo para obtener un movimiento continuo en la dirección especificada.

Tabla 14 Pruebas sensibilidad al movimiento del sistema (horizontal)

Angulo mínimo detectado (grados)	Dirección del movimiento
+6	Derecha
-6	Izquierda
+5,5	Hacia la derecha en equilibrio
-5,8	Hacia la izq. aún en equilibrio

Tabla 15 Pruebas sensibilidad al movimiento del sistema (vertical)

Angulo mínimo detectado (grados)	Dirección del movimiento
+6	Derecha
-6	Izquierda
+5	Hacia la derecha en equilibrio
-5,5	Hacia la izq. aún en equilibrio

Las pruebas de sensibilidad del dispositivo a comenzar un desplazamiento en las direcciones horizontal o vertical fueron satisfactorias ya que existe simetría tanto en el desplazamiento en los dos sentidos como en la relación de desplazamientos verticales y horizontales.

El dispositivo fue calibrado varias veces hasta lograr obtener este resultado. Con esta sensibilidad es posible manipular el dispositivo sin tener que realizar flexiones dolorosas ni levantar demasiado algún extremo de la plataforma para lograr la inclinación.

Otro tipo de prueba fue realizada por varios usuarios. Esta prueba consistía en una aplicación en la que se debía presionar un área en color blanco en un fondo azul (un botón blanco en un fondo azul) que cambia a una ubicación aleatoria cada vez que se logra presionarla.

En la Ilustración 44, se muestra una secuencia normal de la aplicación de prueba.

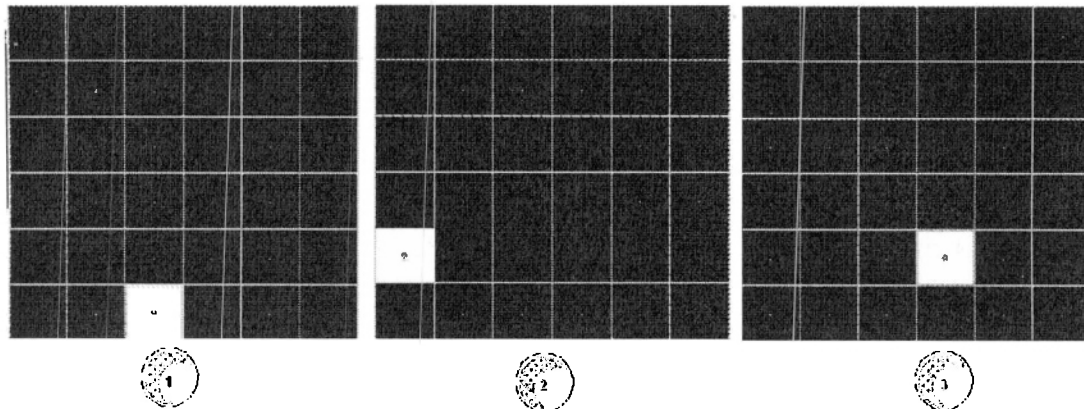


Ilustración 44 Secuencia de la aplicación de prueba

El código de esta aplicación de prueba se encuentra en el anexo. (sección 5.4).

3.6 Problemas encontrados en el desarrollo del dispositivo

Durante el desarrollo del dispositivo se presentaron varios problemas de diferentes índoles. Los problemas se pueden clasificar en dos áreas: los problemas de *software* y los problemas de *hardware*.

Los principales problemas encontrados se listan a continuación:

- Selección del microsistema que cumpla con las características requeridas
 - Seleccionar un microsistema de bajo costo.
 - Que cumpla con la sensibilidad requerida (g/mV).

- Adquisición del microsistema seleccionado
 - Buscar un proveedor con existencias en el país.
 - En su defecto, ordenar suficientes CIs previniendo posibles defectos de fábrica o daños por el usuario.

- Montaje del microsistema
 - Construcción y desarrollo de una superficie adecuada para el montaje.
 - Revisión de las conexiones para evitar corto circuito o errores de conexión de pines.

- Manejo del microsistema
 - Calibración adecuada del sistema de conversión analógica digital.
 - Revisión del diagrama de pines del circuito integrado.

- Sensibilidad a la descarga electrostática
 - Almacenamiento del dispositivo en una bolsa plástica especial para su protección.

- Sensibilidad a la polarización inversa (daño del dispositivo).
 - Daño permanente del dispositivo si se conecta inadecuadamente.

- Problema de adaptación del *driver* USB
 - Ubicación de los pines de entrada del *driver* USB
 - Adecuación de los niveles de voltaje para cada una de las entradas.
 - Los elementos del circuito USB no están disponibles por separado es necesario adquirir un nuevo *mouse*.

4. Conclusiones y trabajo futuro

4.1 Conclusiones

Las pruebas de movimiento realizadas fueron exitosas, por lo tanto, podemos concluir que el *mouse* aquí presentado puede ser utilizado como un sustituto para la navegación e interacción con la computadora en personas con discapacidad motriz en sus extremidades superiores.

Se comprobó que se pueden realizar todas las tareas típicas que se realizan con un *mouse*: navegar en Internet, abrir y cerrar programas, seleccionar menús, desplazar objetos, utilizar programas de dibujo y apagar la computadora apropiadamente, entre otras.

Para la reintegración, asistida por el dispositivo desarrollado en este proyecto, de una persona a la vida laboral será necesario un periodo de entrenamiento en el que podrá familiarizarse con su uso.

El programa de entrenamiento puede ser de gran ayuda para esta etapa en la que el usuario se sensibiliza con el movimiento y puede optar por solicitar una recalibración (por medio del sistema operativo o por un técnico con el cambio de constantes en el microcontrolador).

Un usuario puede emplear el dispositivo con facilidad. Sólo le deben ser indicados con mucha claridad el cómo se realiza la conexión de los módulos, la posición y ángulos máximos que alcanza el dispositivo además de la configuración de parámetros por parte de un técnico para su uso más comfortable.

El funcionamiento del *mouse* fue entendido de manera rápida y sencilla por los sujetos de prueba. La implementación de los movimientos se comprendió con

facilidad y en el caso del uso de botones sucedió lo mismo ya que el un esquema muy similar al del *mouse* para computadora tradicional.

La compatibilidad fue un aspecto que tomado en cuenta desde el comienzo del proyecto. El dispositivo funciona en la mayoría de las computadoras personales que incluyen interfaces USB, incluyendo las de alta y baja velocidad. Por este motivo, el dispositivo funciona también con computadoras no muy actuales con interfaces USB. Esto significa que, tanto en las computadoras más recientes cómo en aquellas con un máximo de ocho años de antigüedad, es posible hacer uso del *mouse* presentado.

Adicionalmente, no se requirió de alimentación externa para el sistema completo. Todas las líneas de alimentación provienen de alguno de los puertos del concentrador USB de la computadora.

Cuando el dispositivo ha sido instalado es posible operarlo sin necesidad de hacer uso de las extremidades superiores. Teniendo el dispositivo su alimentación eléctrica desde el concentrador USB no se requieren baterías, lo cual asegura que el usuario dependerá mínimamente de la asistencia de un tercero.

Algo valioso sobre el dispositivo desarrollado es que se obtuvo conocimiento teórico y técnico sobre dispositivos de interfaz humana y el cómo adaptar nuevas tecnologías para realizar implementaciones cada vez más cómodas, menos dañinas durante su uso prolongado y mucho más intuitivas en cuanto a su manipulación.

Además de los dispositivos de interfaz humana, existe la posibilidad de realizar adaptaciones al dispositivo para que realice funciones útiles en otras ramas. Estas posibles aplicaciones se listan en la sección Trabajo a futuro.

4.2 Trabajo a futuro

Por las características del proyecto, existen aplicaciones alternativas para las cuales se requeriría del diseño de software o del rediseño de la interfaz con el exterior.

Los pasos que seguirían en el desarrollo del proyecto se enlistan a continuación:

- Desarrollar un prototipo precomercial
- Elaboración de pruebas con un prototipo precomercial
- Adaptación del dispositivo para la medición de variables fisiológicas. (monitoreo, rehabilitación e.g.)
- Adaptación del dispositivo para aplicaciones médicas como rehabilitación, monitoreo, detección de enfermedades particulares
- Adaptación del dispositivo para otras aplicaciones como medidor de vibraciones y golpes, para detectar sismos, corregir vibraciones en un disco duro, entre otras

5. Anexo

5.1 Patentes de dispositivos similares

5.1.1 Sistema y método para controlar automáticamente el movimiento de información en la pantalla.

Número de patente: 6351273

Fecha: Feb 26, 2002

Inventor: Jerome H. Lemelson, John H. Hiett

El sistema incluye un sensor para monitorear la posición de la cabeza del usuario y el ojo del mismo, además de un algoritmo de activación que utiliza un sistema neural para encontrar las coordenadas que se van a implementar en la computadora (Como se muestra en la Ilustración 45). La función de subir o bajar la información que se muestra en la pantalla se activa a través de las coordenadas obtenidas al moverse el ojo del usuario. La ventaja de este sistema es que permite el libre movimiento de la cabeza mientras se trabaja.²³

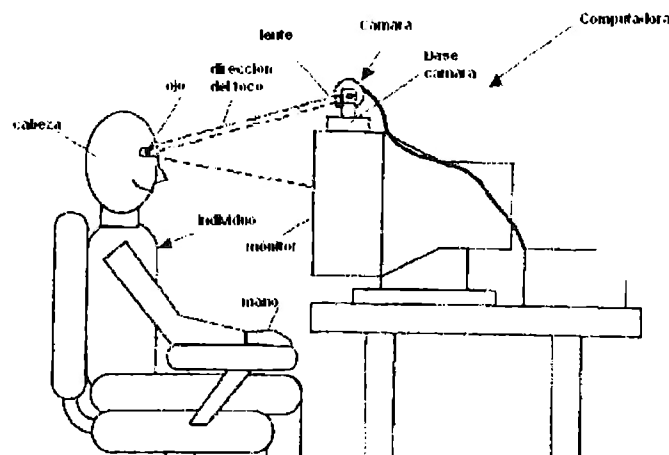


Ilustración 45 Sistema y método para controlar automáticamente el movimiento de información en la pantalla

Fuente: <http://www.google.com/patents?id=34sKAAAAEBAJ>

5.1.2 *Mouse Epoc*

Compañía: Emotiv system

Este dispositivo hace uso del electroencefalograma para detectar patrones de comportamiento cerebral y asociarlos a acciones. Cuenta con 16 sensores que se comunican de forma inalámbrica con la computadora.

El dispositivo se acompaña de una aplicación de entrenamiento para que el usuario sepa de qué forma puede controlarlo. En la Ilustración 46, se muestra un casco con los sensores, su inconveniente es que si este se mueve la obtención de las señales se modifica y las acciones se vuelven confusas, además de que el precio es de 299 dólares.²⁴

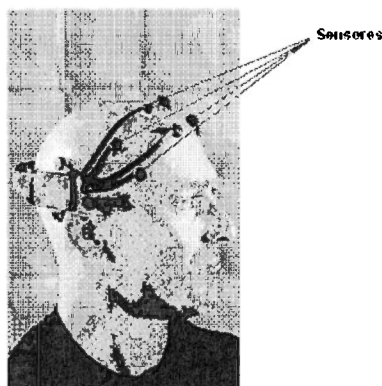


Ilustración 46 *Mouse Epoc*
Fuente: <http://www.businessweek.com/magazine>

5.1.3 Seguidor de la dirección de movimiento de los ojos.

Número de patente: 6152563

Fecha: Nov 28, 2000

Inventores: Thomas E. Hutchinson, Christopher Lankford, Peter Shannon

Este sistema de detección (ver Ilustración 47) usa un diodo de emisión de luz infrarroja montada coaxialmente con un eje óptico y lentes de un sistema de video

sensitivo a infrarrojo para grabar imágenes directamente del ojo del usuario. La luz infrarroja entra en el ojo, se absorbe y es emitida de nuevo por la retina causando un efecto en el cual la pupila brilla más que el resto del ojo. La computadora incluye *hardware* y *software* para adquirir la imagen, digitalizarla y analizarla. Una vez que se obtiene esta información el software calibra el sistema y permite determinar el punto de la pantalla al cual está mirando el usuario.²⁵

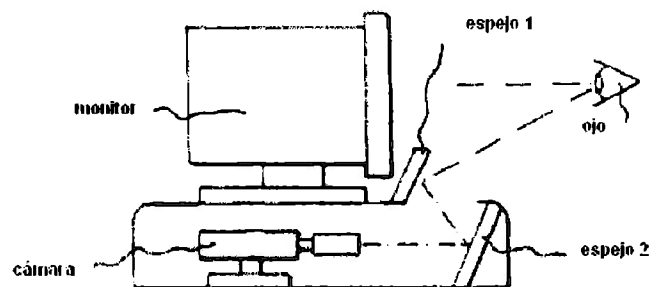


Ilustración 47 Seguidor de la dirección de movimiento de los ojos

Fuente: <http://www.google.com/patents?id=Nm8FAAAAEB AJ>

5.1.4 Método y aparato para controlar el movimiento del cursor

Número de patente: 5367315

Fecha: Nov 22, 1994

Inventor: Shaugan Pan

Compañía: Eyetech Corporation

Es un microprocesador controlado por un sistema de luz infrarroja que indica la posición del ojo y la cabeza en tiempo real. Permite el control del cursor por medio del movimiento de la cabeza y los ojos. El interruptor de inicio puede ser activado con el pie (ver Ilustración 48 Método y aparato para controlar el movimiento del cursor).

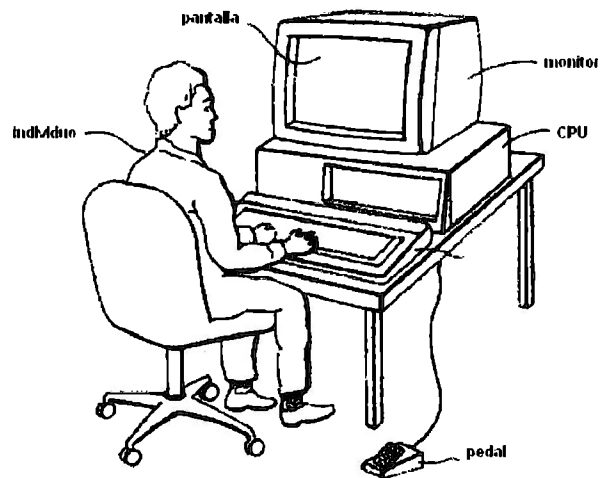


Ilustración 48 Método y aparato para controlar el movimiento del cursor
 Fuente: <http://www.google.com/patents?id=jM4dAAAAEBAJ>

5.2 Programa del dispositivo en el ATMEGA128

```
//Programa de procesamiento de la señal de los acelerómetros
// Para generar los pulsos del mouse
//autores: Rebeca Becerril, Carlos Iván Monroy
```

```
#include <stdio.h>           //librerías a utilizar
#include <sfr_defs.h>
#include <interrupt.h>

//declaración de constantes
//constantes para el movimiento en x
#define EDOX_0 0
#define EDOX_1 2
#define EDOX_2 1
#define EDOX_3 3

//constantes para el movimiento en y
#define EDOY_0 0
#define EDOY_1 2
#define EDOY_2 1
#define EDOY_3 3

//definición de los límites de voltaje del acelerómetro
//horizontal
#define Q_MINX 40
#define Q_EQUX 120
#define Q_MAXX 200
#define Q_ERRX 20

//definición de los límites de voltaje del acelerómetro vertical
#define Q_MINY 40
#define Q_EQUY 120
```

```

#define Q_MAXY 200
#define Q_ERRY 20

unsigned int estadox=EDOX_0;
unsigned int estadoy=EDOY_0;
unsigned char estado_anteriorx; //0-->derecha, 1-->equilibrio, 2-->izquierda
unsigned char estado_anteriory; //0-->arriba, 1-->equilibrio, 2-->abajo

//***** variables globales
unsigned int val_adcx=Q_EQUX;
unsigned int val_adcy=Q_EQUY;
int vert_hor=1; //1 --> horizontal, -1 --> vertical
//***** fin variables globales

//***** interrupciones

//Interrupcion ADC
ISR(ADC_vect){

    if(vert_hor>0){
        vert_hor*=-1;
        ADMUX=(1<<REFS0) | (1<<REFS1) | (1<<ADLAR);

        ADMUX|= (0<<MUX4)|(0<<MUX3)|(0<<MUX2)|(0<<MUX1)|(0<<MUX0);

        //lectura acelerometro vertical
    }else{
        vert_hor *=-1;
        ADMUX=(1<<REFS0) | (1<<REFS1) | (1<<ADLAR);

        ADMUX|= (0<<MUX4)|(0<<MUX3)|(0<<MUX2)|(0<<MUX1)|(1<<MUX0);

    }

    if(vert_hor > 0){
        val_adcx = ADCH;
    }
    else{
        val_adcy = ADCH;
    }

    //movimientos horizontales
    if ((val_adcx < (Q_EQUX-Q_ERRX)) && ((val_adcx >= (Q_MINX))) &&
(estado_anteriorx != 0)){
        retardo();
        izquierda();
        PORTA=0xFE;
        estado_anteriorx=2;
    }
}

```

```

        }else if((val_adcX >= (Q_EQUX-Q_ERRX)) && (val_adcX <=
(Q_EQUX+Q_ERRX))){
            estado_anteriorx=1;
            PORTA=0xFD;
        }else if((val_adcX > (Q_EQUX+Q_ERRX)) && ((val_adcX <=
(Q_MAXX))) && (estado_anteriorx != 2)){
            retardo();
            derecha();
            PORTA=0xFB;
            estado_anteriorx=0;
        }else{
            }

        //movimietnos verticales
        if ((val_adcY < (Q_EQUY-Q_ERRY)) && ((val_adcY >= (Q_MINY))) &&
(estado_anteriorY != 0)){
            retardo();
            abajo();
            PORTC=0xFE;
            estado_anteriorY=2;
        }else if((val_adcY >= (Q_EQUY-Q_ERRY)) && (val_adcY <=
(Q_EQUY+Q_ERRY))){
            estado_anteriorY=1;
            PORTC=0xFD;
        }else if((val_adcY > (Q_EQUY+Q_ERRY)) && ((val_adcY <=
(Q_MAXY))) && estado_anteriorY != 2){
            retardo();
            arriba();
            PORTC=0xFB;
            estado_anteriorY=0;
        }else{
            }

        ADCSRA|=(1<<ADSC);
                                                    //RE-Start Conversion.
    }

```

//***** fin interrupciones

int main(void)

{

```

    DDRA = 0xFF;           //OUT
    PORTA= 0xFF;
    DDRB = 0xFF;
    PORTB= estadox;       //salida a driver mouse
    DDRC = 0xFF;         //PORTA todo es salida
    PORTC= 0xFF;
    DDRE = 0xFF;

```

```

PORTE= estadoy;                //salida a driver mouse
DDRF = 0x00;
PORTF= 0xFF;

//Configuracion del ADC

ADMUX=(1<<REFS0) | (1<<REFS1) | (1<<ADLAR);

ADMUX|= (0<<MUX4)|(0<<MUX3)|(0<<MUX2)|(0<<MUX1)|(0<<MUX0);

ADCSRA=(1<<ADPS2)|(0<<ADPS1)|(1<<ADPS0);    // ADPS=101, prescaler
32, FCLKADC=125kHz

ADCSRA|=(1<<ADIE);                //ADC
Interrupt MAsk bit, enable interrupt at end of adconversion

ADCSRA|=(1<<ADEN);                //ADC enable
ADC

sei();
ADCSRA|=(1<<ADSC);
//inicio del programa

while (1){

}

}

//***** funciones

//-----retardo normal-----
void retardo(){

    int i,j,k;

    //    for( i = 0; i < 5; i++){
        for( j = 0; j < 1000; j++){
            for( k = 0; k < 100; k++){
                DDRD = DDRD;
            }
        }
    //}

}

//-----movimiento a la derecha-----
void derecha(){

    if(estadox==EDOX_0){
        estadox=EDOX_2;
    }else if(estadox==EDOX_1){
        estadox=EDOX_0;
    }
}

```

```

        }else if(estadox==EDOX_2){
            estadox=EDOX_3;
        }else if(estadox==EDOX_3){
            estadox=EDOX_1;
        }else{
        }
    PORTB= estadox;
}

//-----movimiento a la izquierda-----
void izquierda(){

    if(estadox==EDOX_0){
        estadox=EDOX_1;
    }else if(estadox==EDOX_1){
        estadox=EDOX_3;
    }else if(estadox==EDOX_2){
        estadox=EDOX_0;
    }else if(estadox==EDOX_3){
        estadox=EDOX_2;
    }else{
    }

    PORTB= estadox;
}

//-----movimiento hacia arriba-----
void arriba(){

    if(estadoy==EDOY_0){
        estadoy=EDOY_2;
    }else if(estadoy==EDOY_1){
        estadoy=EDOY_0;
    }else if(estadoy==EDOY_2){
        estadoy=EDOY_3;
    }else if(estadoy==EDOY_3){
        estadoy=EDOY_1;
    }else{
    }

    PORTE= estadoy;
}

//-----movimiento hacia abajo-----
void abajo(){

    if(estadoy==EDOY_0){
        estadoy=EDOY_1;
    }else if(estadoy==EDOY_1){

```

```
        estadoy=EDOY_3;
    }else if(estadoy==EDOY_2){
        estadoy=EDOY_0;
    }else if(estadoy==EDOY_3){
        estadoy=EDOY_2;
    }else{
        }
    PORTE= estadoy;
}
//***** fin funciones
//fin programa
```

5.3 Notas de aplicación de acelerómetros²⁶

5.3.1 Aceleración rotacional

La aceleración rotacional es otra medida de movimiento que puede ser obtenida por medio de los acelerómetros. Las aplicaciones que pueden derivarse de esto son para determinar la carga desbalanceada y la compensación rotacional para las lavadoras o estabilidad en las cámaras de video.

El desbalance de la carga puede ser determinado por el hecho de medir la vibración, pero es más eficiente para prevenir el desbalance antes de que suceda. El acelerómetro monitorea la órbita rotacional de la, por medio de las revoluciones por minuto y la geometría elíptica calcula y predice un desbalance de cargas. Un programa puede monitorear la órbita rotacional y desplegar las revoluciones por minuto y la elíptica de manera gráfica, como se muestra en la Ilustración 49 .

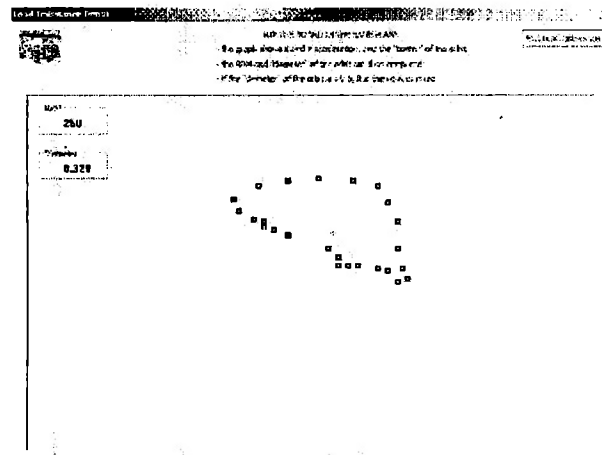


Ilustración 49 Programa que determina el desbalance de cargas en una lavadora
<http://www.freescale.com/>

5.3.2 Medición de vibraciones

Con este tipo de medición es posible detectar actividad sísmica y es útil para la detección de temblores, mantenimiento preventivo de motores, corrección de vibración en discos duros, medición de acústica y control de la misma, entre otros.

El mantenimiento preventivo de motores puede demostrarse haciendo uso de un módulo de análisis de vibraciones que se muestra en la Ilustración 50.

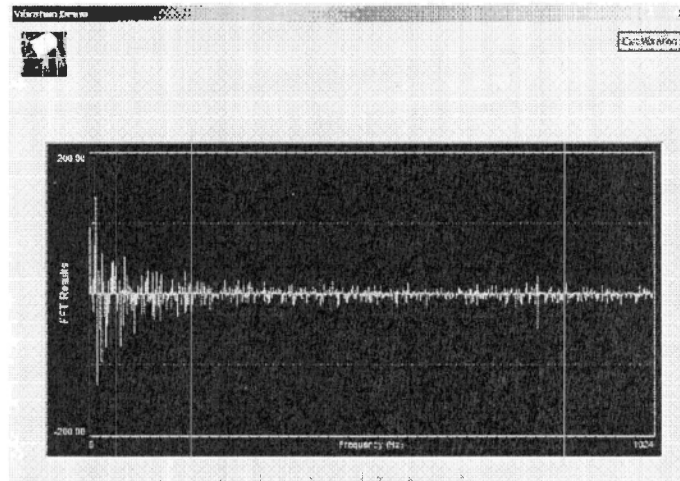


Ilustración 50 Interfaz de monitoreo de vibración
<http://www.freescale.com/>

La resonancia estructural en la maquinaria incrementa generalmente el nivel de vibración lo que lleva a una falla prematura. Un acelerómetro puede predecir la falla reconociendo la huella de la vibración, ya que cada movimiento tiene una huella de vibración que comprende varios niveles de vibración armónica. El acelerómetro es capaz de determinar los armónicos del motor y monitorear cuándo los valores cambiaron, prediciendo un problema con la maquinaria antes que ocurra una falla.

Los acelerómetros pueden ser usados también para detectar la vibración rotacional de los discos duros. Cuando las características de un disco duro operacional se vuelven muy grandes o contienen cierto contenido en el espectro, el desempeño del disco duro se ve comprometido: errores de lectura/escritura, tiempos de búsqueda elevados, pérdida de información. El acelerómetro detecta estas vibraciones y genera una señal para que el disco duro se ajuste para una mejor operación.

5.3.3 Mediciones de golpes

Las aplicaciones de este tipo de medición van desde almacenamiento de datos en las cajas negras, grabación de eventos, protección de discos duros, monitoreo de manejo de paquetes para determinar tipo de manejo durante el transporte (productos frágiles, etc.)

El módulo de detección de golpes provee una demostración para usar el acelerómetro no sólo para detectar que ocurrió un golpe, sino también para indicar el eje en el que ocurrió. El ejemplo emite un *beep* al momento en que hay un golpe en el eje X, emite dos *beeps* cuando fue en el eje Y y tres *beeps* cuando fue en el eje Z. Esta aplicación puede ser ampliada agregando un programa que muestree la señal del golpe y que sea capaz de realizar diferentes acciones para los distintos tipos de golpe. Debido a que la placa TRIAX está diseñada para acelerómetros de $\pm 1.5g$, un golpe se detecta siendo mayor de 1g.

5.3.4 Descripción de la placa TRIAX

Es usada para demostrar diferentes aplicaciones, tales como las que se hablaron con anterioridad. Los componentes básicos son tres acelerómetros, un microcontrolador, comunicación vía puerto serie, EEPROM para recolección de datos y un *buzzer*. El microcontrolador usado en la placa de pruebas es el MC68908KX8, que fue seleccionado debido a que tiene un SCI que es requerido para la comunicación serie, cuatro canales ADC (se usan tres para las salidas de los acelerómetros) y una memoria FLASH de 8Kb que se usa para la calibración de los datos y "recordar" qué módulo fue el último que se usó.

5.4 Código de la aplicación de prueba

```
import java.net.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.io.*;

public class Botones extends JFrame {

    JButton [][] botones = new JButton[6][6];
    public int botx = 0;
    public int boty = 0;

    boolean no_cambiar = true;

    public Botones( ){
        super( "Entrenar" );
        //nickname = JOptionPane.showInputDialog(null,"escriba su nickname");
        GUI();
        initListener();

        System.out.println("entrenar con botones");

        while (true){

            //Random r = new Random();
            int botx = (int) (Math.random()*6);
            int boty = (int) (Math.random()*6);

            System.out.println("x: "+ botx + " y:"+ boty);

            botones[botx][boty].setEnabled(true);
            botones[botx][boty].setText("O");
            botones[botx][boty].setBackground(Color.WHITE);

            while(no_cambiar){
                //espera hasta que el boton se haya presionado
            }

            botones[botx][boty].setEnabled(false);
            botones[botx][boty].setText(".");
            botones[botx][boty].setBackground(Color.BLUE);
            botones[botx][boty].setFocusable(false);
            botones[botx][boty].setEnabled(false);
            System.out.println("x1: "+ botx + " y1:"+ boty);
        }
    }
}
```

```

        no_cambiar = true;
        System.out.println("salio del no cambiar");

    }//while main true
}

public static void main( String argv[] ) throws Exception{
    Botones ventana = new Botones( );

    //System.exit(0);

}//fin main

public void GUI(){
    Container ContentPane = getContentPane();
    ContentPane.setLayout( new BorderLayout() );

    JPanel centro = new JPanel(new GridLayout( 6,6 ));
    for (int i = 0; i < 6; i++){
        for (int j = 0; j < 6; j++){
            botones[i][j] = new JButton(".");
            botones[i][j].setEnabled(false);
            botones[i][j].setBackground(Color.BLUE);
            centro.add(botones[i][j]);
        }
    }

    ContentPane.add( centro, BorderLayout.CENTER );
    /* */
    setSize( 600,600 );
    setVisible( true );
    setResizable( false );
    setDefaultCloseOperation( EXIT_ON_CLOSE );
    /* */
}//fin GUI

public void initListener(){
    for(int i = 0; i < 6; i++){
        for(int j = 0; j < 6; j++){
            botones[i][j].addActionListener( new ActionListener()
            {
                public void actionPerformed(ActionEvent e)
                {
                    //aqui la acción
                    accion();
                }
            }
        )
    }
}

```


Las interrupciones y subrutinas, almacenan de manera automática la dirección de retorno *Programm Counter* (PC).

Las banderas de registros son:

- *Carry* (C)
- *Zero* (Z)
- *Negative* (N)
- *Overflow Complemento a Dos* (V)
- *Sign* ($S=N \oplus V$)
- *Half Carry* (H)
- *Bit Copy Storage* (T)
- *Global Interrupt Enable* (I)

Cada instrucción del microcontrolador mide 16 ó 32 bits, es por esta razón que la *Flash* se encuentra organizada en bloques de 16 bits, para alojar 64K de instrucciones. El *PC* tiene una longitud de 16 bits para poder localizar las 64K (1024Kbits/16) de direcciones.

La memoria se encuentra dividida en dos secciones:

- *Aplication Flash*
- *Boot Flash*

Esta memoria se puede programar por 3 vías:

- *SPI (Serial Programming Interface)*
- *JTAG (Join Test Action Group IEEE1149.1, Standard Test Access Port and Boundary-Scan Architecture)*
- Paralelo

Para la localizar datos en la memoria de programa, es necesario utilizar las instrucciones *SPM*, *LPM (Load Programm Memory)* y *ELPM (Extended Programm Memory)* utilizando el registro Z.

El ATmega128 tiene 4096 bytes de SRAM interna. Los 32 Registros están mapeados dentro de esta memoria en las direcciones \$0000 - \$001F. La memoria de I/O de la \$0020 a la \$005F. Para este microcontrolador, existe una zona de memoria adicional de I/O que comprende las desde la dirección \$0060 hasta la \$00FF.

La EEPROM se encuentra en un espacio separado de memoria, en cual puede ser leída o escrita en *bytes*. Cuando la EEPROM es leída, el CPU es detenido cuatro ciclos de reloj antes de que la siguiente instrucción se ejecutada. Para la operación de escritura, el CPU es detenido 2 ciclos de reloj.

La lectura o escritura de la EEPROM se realiza mediante el uso de los registros:

- EEDR (*DATA REGISTER*). Para la operación de lectura, es aquí donde se almacena el valor leído de la memoria. Para las operaciones de escritura, es aquí donde reside el valor a escribir en la memoria.
- EEAR (*ADDRESS REGISTER*), para el ATmega128 y debido a la cantidad de memoria EEPROM disponible, este registro se separa en EEARH y EEARL, que en conjunto, especifican la dirección de memoria de la EEPROM que será leída o escrita.
- EECR (*CONTROL REGISTER*). Actualmente este registro cuenta con cuatro banderas de estado direccionables en bits.
 - EERIE (*EE Ready Interrupt Enable*). Habilita la interrupción
 - EEMWE (*EE Master Write Enable*)
 - EEWE (*EE Write Enable*)
 - EERE (*EE Read Enable*)

El microcontrolador tiene la opción de seleccionar varias fuentes de reloj, las cuales son:

- Resonador Externo Cristal/Cerámico
- Cristal externo de baja frecuencia
- Oscilador RC Externo
- Oscilador RC calibrado Interno

- Reloj Externo (Generador de funciones u otro sistema microcontrolado).

Las diferentes fuentes de reloj son seleccionadas mediante la configuración de "fusibles". El uso del Oscilador RC Calibrado Interno, permite que el sistema no necesite de algún dispositivo externo para funcionar. Cuando el CPU "despierta" debido a una baja de voltaje o modo de ahorro de energía, el reloj-fuente seleccionado da el tiempo de arranque del CPU, para permitir la operación estable del oscilador.

Cuando el MCU inicia desde *reset*, existe un tiempo adicional para asegurar que el voltaje tenga su nivel adecuado. La selección del tiempo de retardo para alcanzar VCC, se especifica mediante la programación de los fusibles SUT (*Start-Up Time*).

La configuración de los fusibles SUT, depende del tipo oscilador configurado mediante los fusibles CLKSEL. De fabrica, los microcontroladores están configurados en CLKSEL="0001" y SUT="10", es decir Oscilador Interno RC calibrado 1MHz, *start-up time*: $6Ck + 64$ ms.

Cabe destacar que:

- Ciclo reloj se obtiene haciendo $1/\text{Frecuencia del cristal}$.
- Ciclo de Máquina. Es el número de ciclos de reloj necesarios para traer una instrucción, decodificarla, ejecutarla, y de ser el caso almacenar el resultado en memoria externa.
- Ciclo de instrucción. Es el número de ciclos de maquina necesarios para ejecutar una instrucción. En algunos sistemas con alto desempeño (gracias al *pipeline*), el ciclo de instrucción es igual al ciclo de reloj

La mayoría de las instrucciones tienen acceso directo a estos registros y la mayoría de estas se ejecutan en un ciclo de reloj. Cada registro tiene asignada una dirección de memoria dentro de las primeras posiciones de la memoria de datos del usuario (SRAM).

El *Stack Pointer* (SP) contiene la dirección de memoria dentro de la *SRAM*, donde se extraen los datos de la pila. Y siempre apunta a la parte superior de la pila. La pila de datos normalmente se utiliza para almacenar direcciones de retorno (RET ADD) en las llamadas a subrutinas e interrupciones, para almacenar variables locales y valores temporales. Cuando se almacena un valor en la pila (PUSH, CALL, INT) provoca un decremento en el valor de $SP = SP - 1$. Cuando se extrae un valor de la pila (POP, IRET, RET) el valor de SP se incrementa $SP = SP + 1$. El SP de los AVR es de 16 bits, especificado como dos registros de 8 bits cada uno SPH y SPL.

Todos los puertos *I/O* de la familia AVR tienen funcionalidad "*Read-Modify-Write*". Esto significa que es posible cambiar la dirección (entrada-salida) de un solo pin de un puerto sin cambiar ni por un momento, el estado de los demás pines relacionados en el puerto de *I/O*. Cualquier pin tiene la energía suficiente para encender un diodo emisor de luz. Todos los pines tienen un sistema de PULL-UP que puede ser deshabilitado individualmente. Para el control de puertos de entrada y salida en la familia AVR existen tres Registros DDRx, PORTx y PINx, siendo x el nombre del puerto al que se quiere acceder (A, B, C, ...). Y una bandera en el registro SFIOR (*Special Function Input Output Registers*) que controla el sistema de PULL-UP general (PUD. *Pull Up Disable*).

La mayoría de los pines de *I/O* se encuentran multiplexados con otros módulos del sistema AVR que le dan mayor flexibilidad a cada pin. Cada pin en ocasiones puede tener hasta 4 módulos asignados en él mismo. Al momento de inicial el MCU el valor de la bandera PUD en el registro SFIOR es igual a cero, es decir que el sistema de PUD se encuentra activado.

DDRx es llamado registro de dirección de puerto. Define la dirección de cada uno de los pines del puerto (x), como entrada (0) o salida (1).

Para poner un puerto como salida, es necesario que PORTxn de un puerto sea enviado a 1 lógico, cuando el pin está configurado como entrada, es activado un resistor interno de PULL-UP.

Para desactivar el sistema de PULL- UP, es necesario escribir en PORTxn un cero lógico, cuando DDRxn=0.

6. Referencias y bibliografía

- ¹ Rehabilitación de la Mano. (s.f.). Recuperado el 4 de Noviembre de 2008, de <http://www.rehabilitaciondelamano.com/joomla/>
- ² Escuela de Medicina. Pontificia Universidad Católica de Chile. (s.f.). LESIONES TRAUMATICAS DE LOS TENDONES. Recuperado el 24 de Octubre de 2008, de http://escuela.med.puc.cl/paginas/publicaciones/textotraumatologia/Trau_Secc05/Trau_Sec05_03.html
- ³ Navarro Díaz, C., & García Domínguez, J. (s.f.). Enfermedad de Dupuytren. Recuperado el 24 de Octubre de 2008, de <http://www.secrepre.org/documentos%20manual%2064.html>
- ⁴ Real Academia Española. (2001). Diccionario de la lengua española - Vigésima segunda edición. Recuperado el 3 de Septiembre de 2008, de http://buscon.rae.es/draeI/SrvltConsulta?TIPO_BUS=3&LEMA=discapacitado
- ⁵ Asociación de Ciegos de Viña del Mar. (s.f.). Para la OMS, discapacidad es "Cualquier restricción o carencia" (resultado de una deficiencia). Recuperado el 3 de Septiembre de 2008, de http://www.geocities.com/asocciegos/para_oms.html?200818?200824
- ⁶ Parra Dussán, Carlos. *Derechos humanos y discapacidad. Marco conceptual*. Universidad del Rosario.
- ⁷ Amate, Esther Alicia y Vásquez. *Discapacidad lo que todos debemos saber*. Pan American Health Org.
- ⁸ INEGI. (s.f.). Consulta Dinámica de Cuadros Estadísticos. Recuperado el 2 de Septiembre de 2008, de [http://www.inegi.gob.mx/est/librerias/leetable.asp?whichpage=1&pagesize=20&cuadro=vi16&columna\(1\)=1&columna\(2\)=2&columna\(3\)=3&columna\(4\)=4&categorias\(1\)=Entidad+Federativa&categorias\(2\)=Tama%F1o+de+Localidad&categorias\(3\)=Bienes+en+la+Vivienda&tipo=1&cnt_](http://www.inegi.gob.mx/est/librerias/leetable.asp?whichpage=1&pagesize=20&cuadro=vi16&columna(1)=1&columna(2)=2&columna(3)=3&columna(4)=4&categorias(1)=Entidad+Federativa&categorias(2)=Tama%F1o+de+Localidad&categorias(3)=Bienes+en+la+Vivienda&tipo=1&cnt_)
- ⁹ Device Class Information for Human Interface Devices (HID). (2001). Recuperado el 24 de Octubre de 2008, de http://www.usb.org/developers/devclass_docs/HID1_11.pdf
- ¹⁰ Device Class Information for Human Interface Devices (HID). (2001). Recuperado el 24 de Octubre de 2008, de

http://www.usb.org/developers/devclass_docs/HID1_11.pdf

¹¹ How Stuff Works. (s.f.). HowStuffWorks "How Computer Mice Work". Recuperado el 28 de Septiembre de 2008, de <http://computer.howstuffworks.com/mouse.htm>

¹² Torrens, R. (2002). 4QD: Mice: how do they work. Recuperado el 30 de Septiembre de 2008, de <http://www.4qdtec.com/meece.html>

¹³ Microchip. (2002). Implementing a Simple Serial Mouse Controller. Recuperado el 30 de Septiembre de 2008, de <http://ww1.microchip.com/downloads/en/AppNotes/00519c.pdf>

¹⁴ Hyde, J. (1999). USB Design by Example: A Practical Guide to Building I/O Devices. John Wiley & Sons.

¹⁵ HP. (19 de Septiembre de 2007). INTEL: Industry Leaders Develop Superspeed USB Interconnect; Popular USB Computer Connection Technology Expands performance with Proposed USB 3.0 Specification. Recuperado el 20 de Octubre de 2008, de <http://0-proquest.umi.com.millennium.itesm.mx/pqdlink?index=12&did=1337984151&SrchMode=1&sid=1&Fmt=3&VInst=PROD&VType=PQD&RQT=309&VName=PQD&TS=1191981207&clientId=23693>

¹⁶ USB.org. (s.f.). *USB.org - Hi-Speed USB*. Recuperado el 24 de Octubre de 2008, de <http://www.usb.org/developers/usb20/>

¹⁷ Freescale Semiconductor, Inc. (2008). MEMS Technology. Recuperado el 7 de Noviembre de 2008, de http://www.freescale.com/files/sensors/doc/fact_sheet/MEMSFS.pdf?fsp=1&WT_TYPE=FactSheets&WT_VENDOR=FREESCALE&WT_FILE_FORMAT=pdf&WT_ASSET=Documentation

¹⁸ Atmel. (2008). 8-bit AVR Microcontroller 128K Bytes In-System Programmable Flash. Recuperado el 20 de Septiembre de 2008, de http://www.atmel.com/dyn/resources/prod_documents/doc2467.pdf

¹⁹ Freescale Semiconductor, Inc. (Mayo de 2005). Using an Accelerometer Evaluation Board. Recuperado el 18 de Noviembre de 2008, de http://www.freescale.com/files/sensors/doc/app_note/AN1925.pdf?fsrch=1

²⁰ Freescale Semiconductor, Inc. (2006). $\pm 1.5g$ X-Axis Micromachined. Recuperado el 18 de Septiembre de 2008, de http://www.freescale.com/files/sensors/doc/data_sheet/MMA2260D.pdf?fsrch=1

²¹ Freescale Semiconductor, Inc. (2006). $\pm 1.5g$ X-Axis Micromachined. Recuperado el 18 de Septiembre de 2008, de http://www.freescale.com/files/sensors/doc/data_sheet/MMA2260D.pdf?fsrch=1

²² Taiwan Turnkey Project Association. (s.f.). Planta de produccion de laminas de espuma de EVA y sandalias de playa. Recuperado el 20 de Noviembre de 2008, de <http://turnkey.taiwantrade.com.tw/showpage.asp?subid=113&fdname=RUBBER+%26+PLASTICS&pagename=Planta+de+produccion+de+laminas+de+espuma+de+EVA+y+sandalias+de+playa>

²³ Lemelson, J., & Hiett, J. (2002). Patente n° US 6,351,273B1. Estados Unidos.

²⁴ Edwards, C. (30 de Julio de 2008). Emotiv's New Mind-Control Headset for PCs. Recuperado el 1 de Octubre de 2008, de http://www.businessweek.com/magazine/content/08_32/b4095000909813.htm?globalbiz+product+review

²⁵ Hutchinson, T., Shannon, P., & Lankford, C. (1998). Patente n° 6,152 563. Estados Unidos.

²⁶ Freescale Semiconductor, Inc. (2008). MEMS Technology. Recuperado el 7 de Noviembre de 2008, de http://www.freescale.com/files/sensors/doc/fact_sheet/MEMSFS.pdf?fsp=1&WT_TYPE=FactSheets&WT_VENDOR=FREESCALE&WT_FILE_FORMAT=pdf&WT_ASSET=Documentation



**TECNOLÓGICO
DE MONTERREY.**

Campus Ciudad de México
División de Ingeniería y Arquitectura
Departamento de Ingeniería
Electrónica y Mecánica

Proyectos de Ingeniería Computacional
Ingeniería en Sistemas Electrónicos

Equipo: Rebeca Zoraya Becerril Fernández
Carlos Iván Monroy Badillo

A00969997
A01109187

Asesores: Dr. Martín Rogelio Bustamante Bello
Dr. José Ramón Álvarez Bada

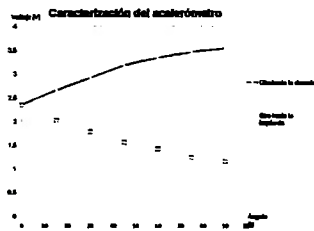
MOUSE PARA PERSONAS CON POCA O NULA MOVILIDAD EN LAS MANOS

Objetivos

- ❖ Diseño de un dispositivo de interfaz humana que sustituya al *mouse* convencional, eliminando el uso de las extremidades superiores.
- ❖ Lograr la compatibilidad con la mayoría de las PCs.
- ❖ Obtener un prototipo experimental seguro y confiable

Desarrollo de los bloques del sistema

1. Diseño de un sistema de apuntador mediante el uso de acelerómetros
2. Se investigó el funcionamiento del *mouse* y se determinaron las señales de entrada y salida
3. Se programó un microcontrolador para simular las entradas del *mouse*; se le dejó preparado para recibir y procesar las señales generadas por los microacelerómetros.
4. Se investigaron, adquirieron y caracterizaron dos microacelerómetros (MEMS).
5. Se interconectaron los bloques y los componentes para integrar el sistema



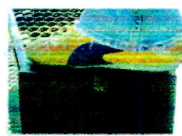
Funcionamiento del sistema integrado

1. Determinación de la inclinación del pie mediante los microacelerómetros.
2. Conversión analógico digital de las señales de los MEMS.
3. Determinación del movimiento por medio del microcontrolador.
4. Envío de señales al driver del mouse.
5. Codificación de la señal mediante el driver del mouse.
6. Interpretación de la señal por medio de la PC.
7. Movimiento del cursor o acción de algún botón

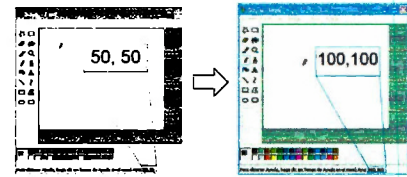
Pruebas y resultados del sistema integrado

- Se logró el acoplamiento de los acelerómetros al microcontrolador; de este dispositivo se obtuvieron las señales necesarias para la operación del *driver* del *mouse*.
- Se logró el funcionamiento del sistema completo en una primera etapa de pruebas de laboratorio.
- Se caracterizó de manera detallada el funcionamiento del sistema, haciendo múltiples movimientos en dos planos.
- Se logró la operación adecuada del sistema en una etapa final en la que se desarrolló el prototipo funcional.

Movimiento del pie y presión de un botón



Acciones en la PC



Conclusiones

- ✓ El prototipo de *mouse* aquí presentado cumple con el objetivo de funcionalidad.
- ✓ Cumple con las características básicas de funcionamiento de los ratones convencionales.
- ✓ No se requirió de alimentación externa para el sistema completo ya que todas las líneas de alimentación provienen de alguna interfaz USB de la computadora.

Trabajo futuro

- Adaptación del dispositivo para aplicaciones médicas como rehabilitación, monitoreo, detección de enfermedades particulares
- Adaptación del dispositivo para otras aplicaciones como medidor de vibraciones y golpes, para detectar sismos, corregir vibraciones en un disco duro, entre otras