

I. Objetivo

Desarrollar una aplicación para Windows capaz de configurar las funciones de un Wiimote, a través de una interfaz gráfica de fácil comprensión; con el propósito de interactuar a distancia con la computadora y facilitar la navegación en aplicaciones con ambientes virtuales.

Además, utilizar el lenguaje C++ a través de WinAPI para crear una aplicación compatible con diversos sistemas operativos Windows, para operar en conjunto con el Wiimote, aprovechando la tecnología inalámbrica Bluetooth y simular las funciones de un ratón y un teclado a distancia.

II. Definición del problema

Hoy en día los ambientes virtuales forman parte de muchas actividades, ya sean académicas, culturales, profesionales o incluso de entretenimiento, y la manipulación de estos ambientes generalmente implica el uso de dispositivos periféricos como un ratón y un teclado en conjunto. Lamentablemente, la movilidad de estos dispositivos se ve limitada por cables de corta distancia y superficies de apoyo que se requieren para su operación adecuada, por lo que en muchas ocasiones estas tareas se consideran muy complicadas, caras o difíciles de implementar, dando por consecuencia que las aplicaciones de ambientes virtuales se descarten de entre diversas opciones, provocando que el usuario prefiera otras formas gráficas distintas a los ambientes virtuales para generar y compartir una fuente de conocimiento que podría ser muy importante y trascendental si se descubriera a través de las tecnologías que involucran la realidad virtual.

Es importante destacar que en muchos sitios, como en las aulas y auditorios donde se llevan a cabo desde clases hasta congresos, se requiere de la manipulación de las computadoras y, cada día más, de aplicaciones con ambientes virtuales, siendo de gran importancia e indispensable contar con la tecnología necesaria para llevar a cabo dichas tareas de manera eficiente y a gran distancia para permitir que tanto alumnos como profesores y profesionales en distintas ramas sean capaces de utilizar sus habilidades libremente y así aprovechar todo su potencial.

Por otro lado, los ratones y los teclados inalámbricos podrían verse como una solución a este problema, sin embargo suelen ser de alto costo y no mejorarían considerablemente la movilidad de los mismos y por ende del usuario, tan sólo incrementarían la distancia a la que podrían funcionar. Por lo tanto, una solución factible para esta problemática sería un dispositivo que cuente con conectividad inalámbrica y que simule las funciones de varios dispositivos periféricos como el ratón y el teclado, además de que no requiera de algún factor físico que limite su posición y movilidad.

Es por eso que, tomando en cuenta la problemática planteada y las características innovadoras del Wiimote, en esta tesis se propone el uso de este dispositivo como alternativa para facilitar la navegación virtual, entre otras funciones.

III. Resumen

En esta tesis se desarrolló la aplicación WiimoteAPI para sistemas operativos Windows, capaz de configurar las funciones de un Wiimote, como son los botones, el acelerómetro y la extensión Nunchuck (tanto sus botones como su joystick). La WiimoteAPI se implementó con el lenguaje de programación C++ a través de WinAPI.

Se aprovechó la tecnología inalámbrica Bluetooth para comunicar el Wiimote con la computadora y se logró simular las funciones de un ratón y un teclado, logrando sustituirlos en el uso de aplicaciones de realidad virtual, así como en otras que requieren la intervención de estos periféricos simultáneamente.

Como ejemplo práctico de la operación del Wiimote y la WiimoteAPI se utilizó la aplicación VMD, comprobando así que la funcionalidad de estos dos componentes permiten asistir y facilitar el aprendizaje de cualquier usuario, tanto de estudiantes en su desarrollo como futuros profesionales en áreas que requieren la navegación virtual, así como de expertos al utilizar aplicaciones como VMD para la manipulación de objetos en ambientes virtuales, o incluso otras.

IV. Antecedentes

2.1 El Wiimote como dispositivo de interfaz humana.

El control remoto de la consola Wii de Nintendo, conocido como **Wiimote**, funciona como el dispositivo de entrada principal de dicha consola. El Wiimote es un dispositivo inalámbrico que utiliza la tecnología Bluetooth para comunicarse con otros dispositivos, utilizando un sistema Broadcom BCM2042, por lo que es posible utilizarlo, a distancia, como dispositivo de entrada de un ordenador; consta de once botones de control factibles para programar y realizar funciones del teclado, un acelerómetro capaz de simular el movimiento del ratón, un sensor de luz infrarroja y un puerto de expansión que permite utilizar otros dispositivos en conjunto con el Wiimote. Cabe destacar que el botón “POWER” del Wiimote únicamente funciona para desconectarlo de cualquier equipo al que está sincronizado y apagarlo.

Uno de los dispositivos a conectar al Wiimote es el Nunchuck, un control de mando o “joystick” que consta de una palanca capaz de moverse 360°, dos botones y un acelerómetro. A continuación se muestran las dos caras del Wiimote y el Nunchuck.



El Wiimote utiliza el protocolo estándar Bluetooth HID (Human Interface Device – Dispositivo de Interfaz Humana) para comunicarse con el dispositivo anfitrión; sin embargo, el Wiimote no utiliza los tipos de datos estándar ni el descriptor HID, el cual escribe los paquetes de datos; por contrario, sólo describe el tamaño de su formato de reporte, dejando el contenido restante indefinido, lo cual permite omitir el uso de controladores.

La comunicación entre el Wiimote y otros dispositivos se da a través de un conjunto de operaciones, las cuales se transmiten por medio de los reportes de entrada y de salida del protocolo HID, transmitiendo así los paquetes de datos distintos que contienen información de los diversos sistemas integrados en el Wiimote, ya sean botones, dispositivos conectados al puerto de expansión, acelerómetros, etc.

2.2 Conexión del Wiimote por Bluetooth.

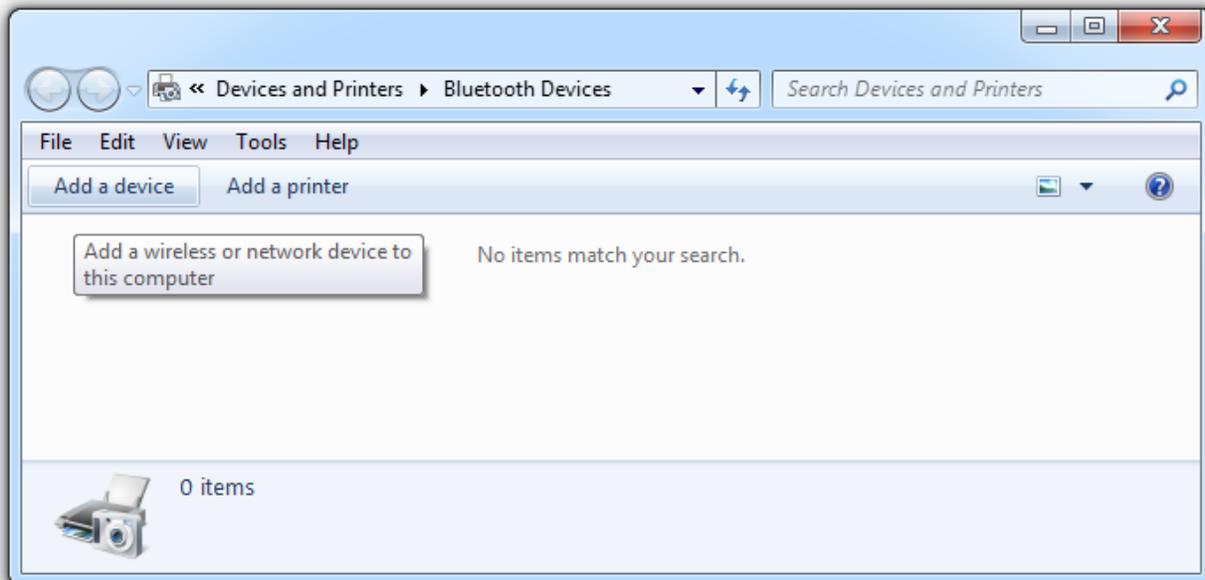
El Wiimote es invocado por medio del SDP (Protocolo de descubrimiento de servicios) de Bluetooth, un protocolo encargado de identificar los dispositivos inalámbricos con el propósito de establecer la comunicación entre estos y el ordenador. El Wiimote reporta la siguiente información a través de este protocolo:

Name	Nintendo RVL-CNT-01
Vendor ID	0x057e
Product ID	0x0306
Major Device Class	1280
Minor Device Class	4
Service Class	0
(Summary of all Class Values)	0x002504

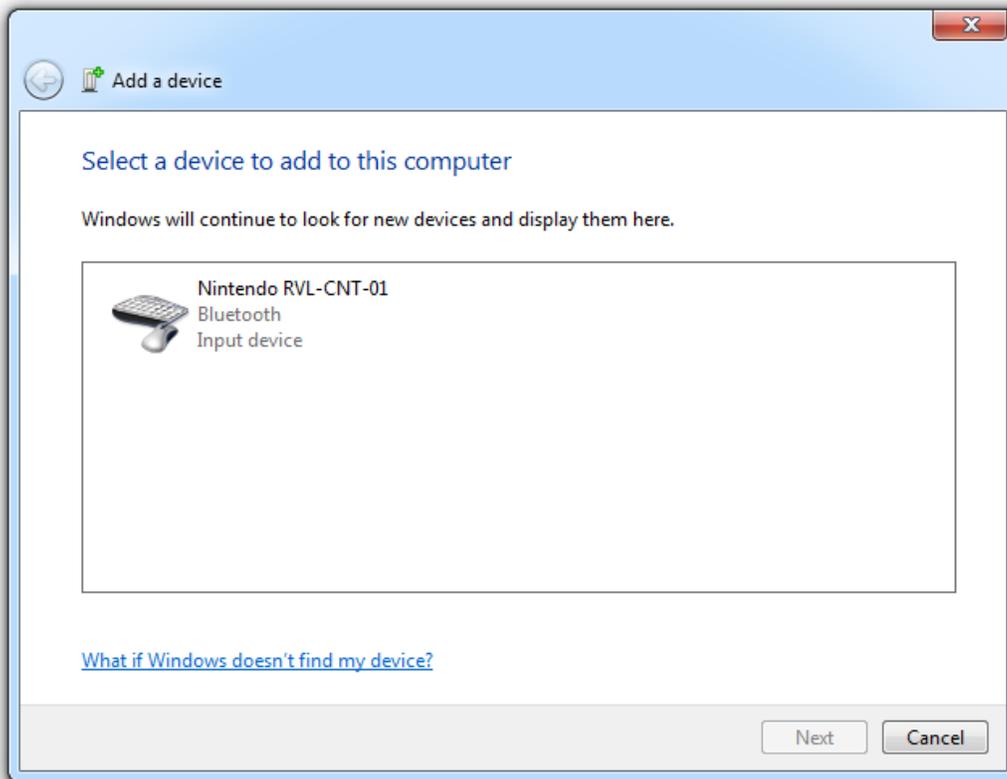
El Wiimote no requiere de autenticación o de algún tipo de cifrado del estándar de Bluetooth para comunicarse. Para reconocerlo desde un ordenador y agregarlo como HID se activa el modo de detección presionando los botones 1 y 2 al mismo tiempo, o presionando el botón de sincronización localizado bajo la cubierta de las baterías. Este estado del Wiimote dura solamente 20 segundos, a menos que se mantengan presionados los botones 1 y 2 para forzar y mantener un estado de detección permanente hasta soltar los botones, a diferencia del botón de sincronización que sólo permite 20 segundos de modo de detección aunque se mantenga presionado. Mientras el Wiimote se encuentra en modo de detección, sus LEDs encienden intermitentemente y el número de LEDs que encienden indica la cantidad proporcional de carga de las baterías.^[14]

A continuación se muestra el proceso de conexión del Wiimote a una computadora:

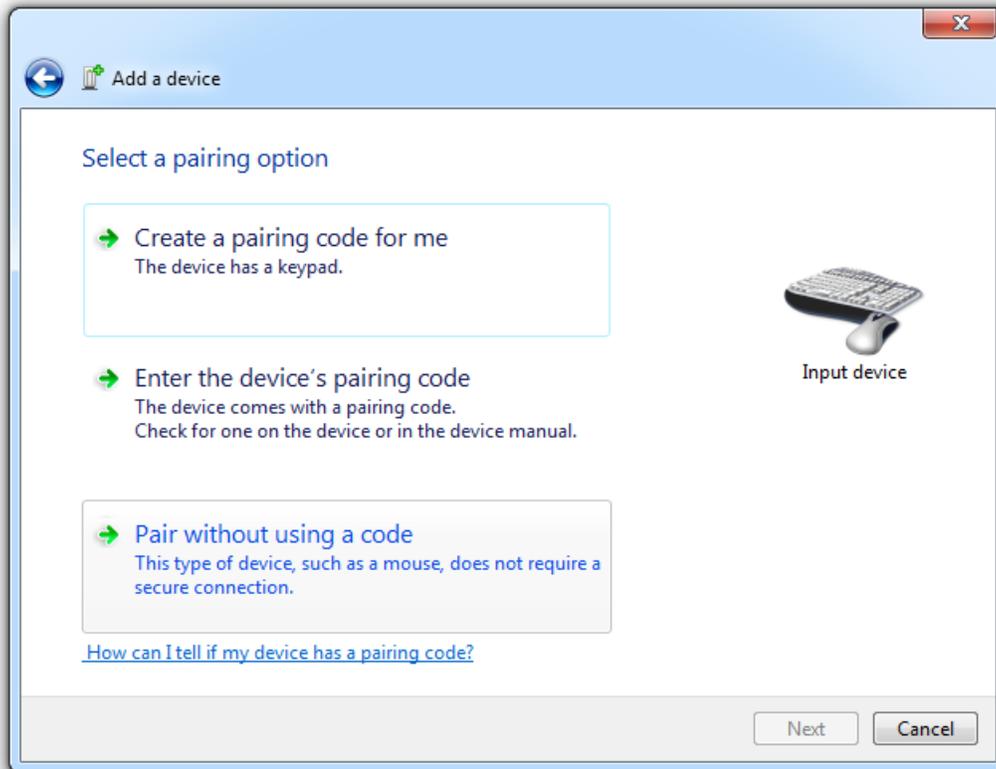
1. Abrir el panel de dispositivos Bluetooth:



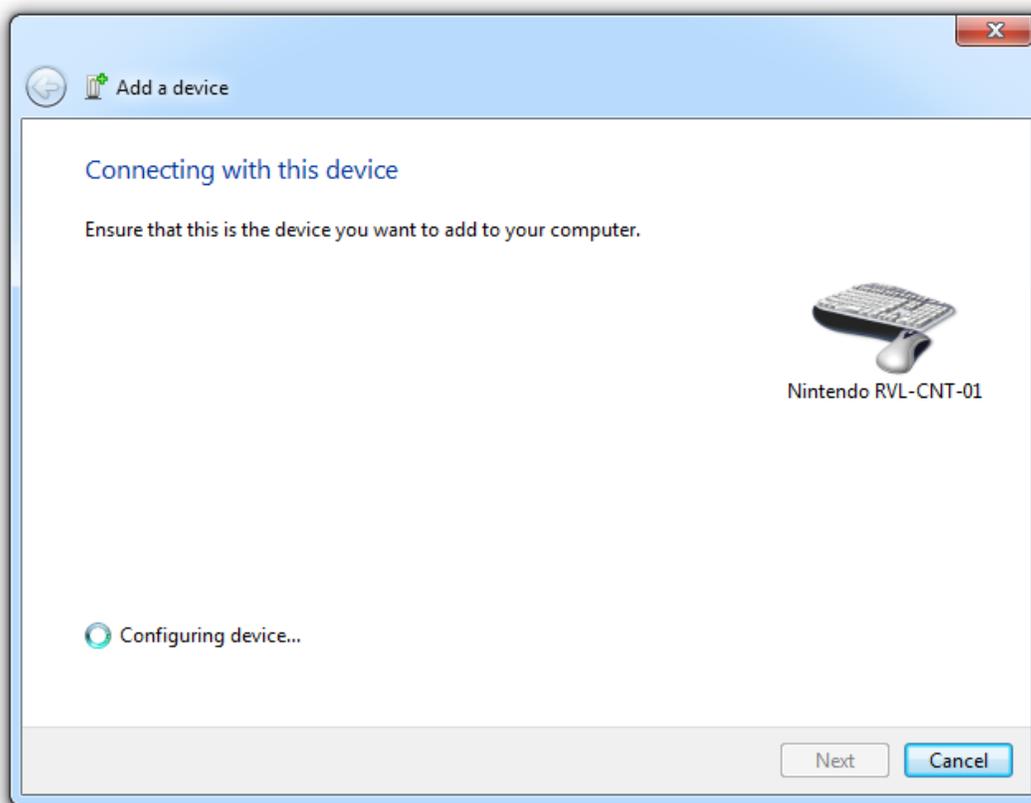
2. Hacer click en “agregar dispositivo” y colocar el Wiimote en modo de detección para visualizar el Wiimote como dispositivo a agregar:



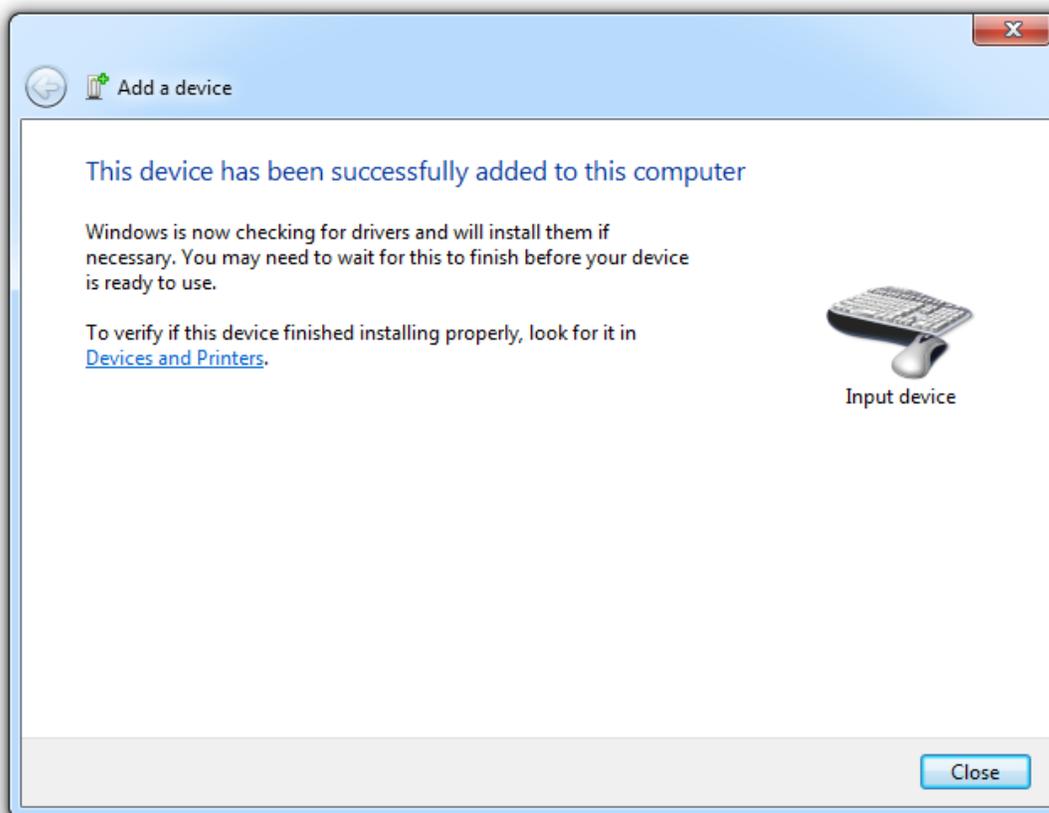
3. Hacer doble click sobre el ícono “Nintendo RVL-CNT-01”, o hacer un click sobre él y hacer click en “siguiente”; a continuación se mostrarán las opciones de conexión para el Wiimote:



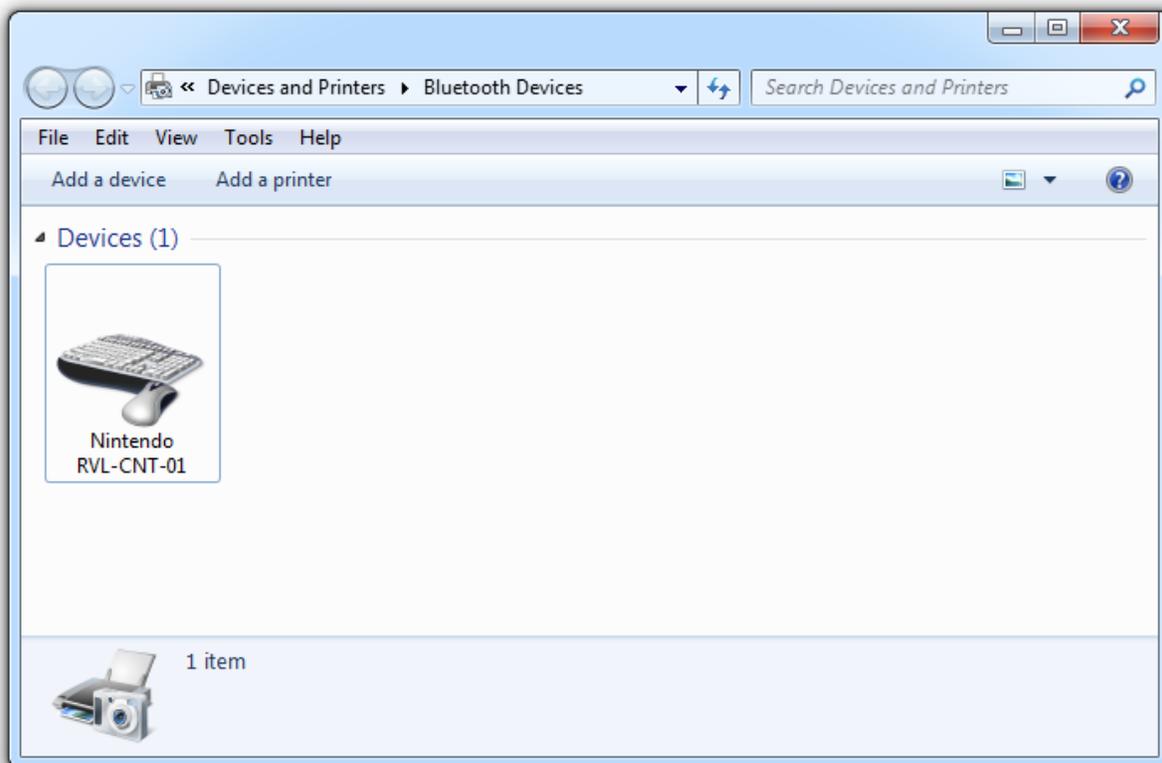
4. Dar click en “Aparear sin código” y esperar a que el equipo configure la conexión con el Wiimote:



5. Una vez que se han sincronizado el Wiimote y la PC con éxito, hacer click en cerrar:



6. Verificar que el Wiimote se agregó correctamente localizándolo en el panel de dispositivos Bluetooth:



7. Una vez que se terminó de utilizar el Wiimote se debe eliminar de los dispositivos Bluetooth ya que es incapaz de recuperar la sincronización después de que la PC o el mismo Wiimote han sido apagados, por lo que este procedimiento se debe realizar cada vez que el equipo o el Wiimote es encendido.

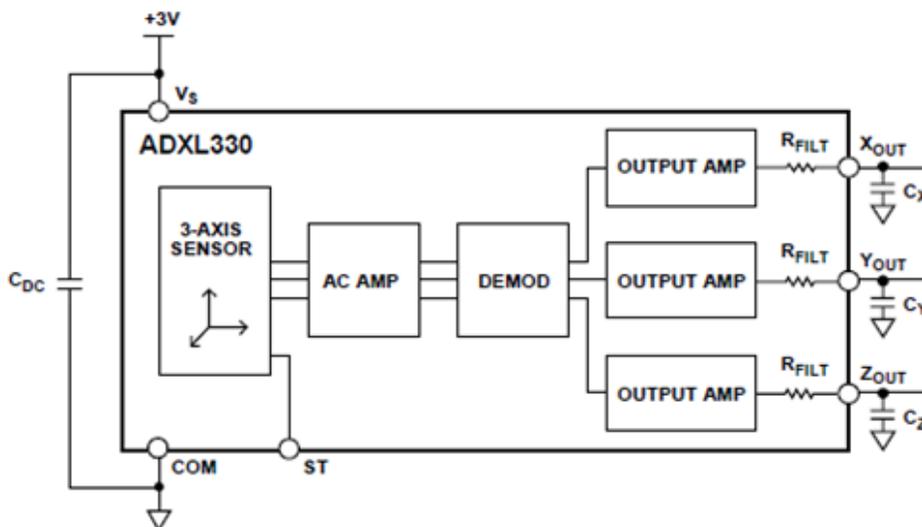
2.3 Acelerómetro ADXL330 en el Wiimote.

Un acelerómetro es un tipo de sensor de movimiento, diseñado para realizar la transducción de un fenómeno físico en una señal fácil de medir. En este caso el fenómeno físico es una fuerza originada por la aceleración a la que es sometido el sensor. Generalmente esta fuerza se convierte a una salida de voltaje linealmente escalada y con cierta sensibilidad.^[5]

El Wiimote cuenta con un acelerómetro ADXL330 de baja potencia, fabricado por Analog Devices, Inc., una de las productoras de sensores inerciales más grandes del mercado, con su proceso bien establecido iMEMS (1993), el cual integró eficazmente el sensor mecánico y el procesamiento de señales en un mismo circuito.^[7]

El ADXL330 ganó el concurso al mejor diseño para el **Wiimote** del Nintendo Wii, estableciendo a los MEMS (Microelectromechanical Systems - Sistemas Microelectromecánicos) como la llave de la innovación en una nueva generación de consolas de videojuegos. Además, este dispositivo MEMS surgió de una mejora del ADXL203, un acelerómetro de dos ejes, añadiendo la detección de movimiento en el eje Z y manteniendo la detección en los ejes X y Y sin cambios esenciales.^[9]

Las aplicaciones más frecuentes del ADXL330 se encuentran en consolas y controles de videojuegos como el Wiimote, dispositivos móviles, dispositivos deportivos y de salud, protección de discos duros y estabilización de imagen. Consta de un sensor de 3 ejes de detección con salidas de voltaje analógico proporcionales a la aceleración registrada, en un solo circuito integrado monolítico. Este sensor mide la aceleración en un rango de ± 3 g y con una sensibilidad del 10%; puede utilizarse en varias aplicaciones que requieran medir la aceleración estática de la gravedad de acuerdo a cierta inclinación a la que es sometido el sensor, así como la aceleración dinámica que resulta de movimientos, impactos y vibraciones.^[5] A continuación se muestra el diagrama de bloques funcional del acelerómetro ADXL330:

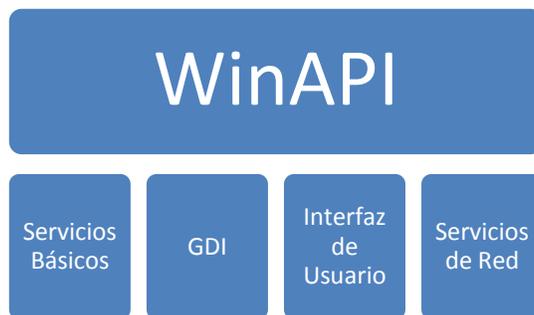


El usuario puede seleccionar el ancho de banda de las señales del acelerómetro usando los pines de salida X_{OUT} , Y_{OUT} y Z_{OUT} , correspondientes a los capacitores C_X , C_Y y C_Z ; y los anchos de banda se pueden configurar para satisfacer la aplicación deseada, con un rango de 0.5 Hz a 1600 Hz para los ejes X y Y, y un rango de 0.5 Hz a 550 Hz para el eje Z. Una vez que las señales salen del sensor, se digitalizan en enteros sin signo de 8 bits.

2.4 API de Windows.

Windows es el sistema operativo más usado en la industria de las computadoras, cubriendo un poco más del 80% del mercado.^[2] Su API, también conocida como **WinAPI**, es un conjunto de *códigos fuente de interfaz* que se utiliza para crear aplicaciones de Windows, con ayuda del SDK (Software Development Kit - Kit de Desarrollo de Software) de Windows, el cual contiene ficheros cabecera, bibliotecas, documentación y otras herramientas. WinAPI está diseñada para los lenguajes de programación C y C++ y es el modo más directo para crear aplicaciones de Windows.

WinAPI consta de cuatro componentes básicos:

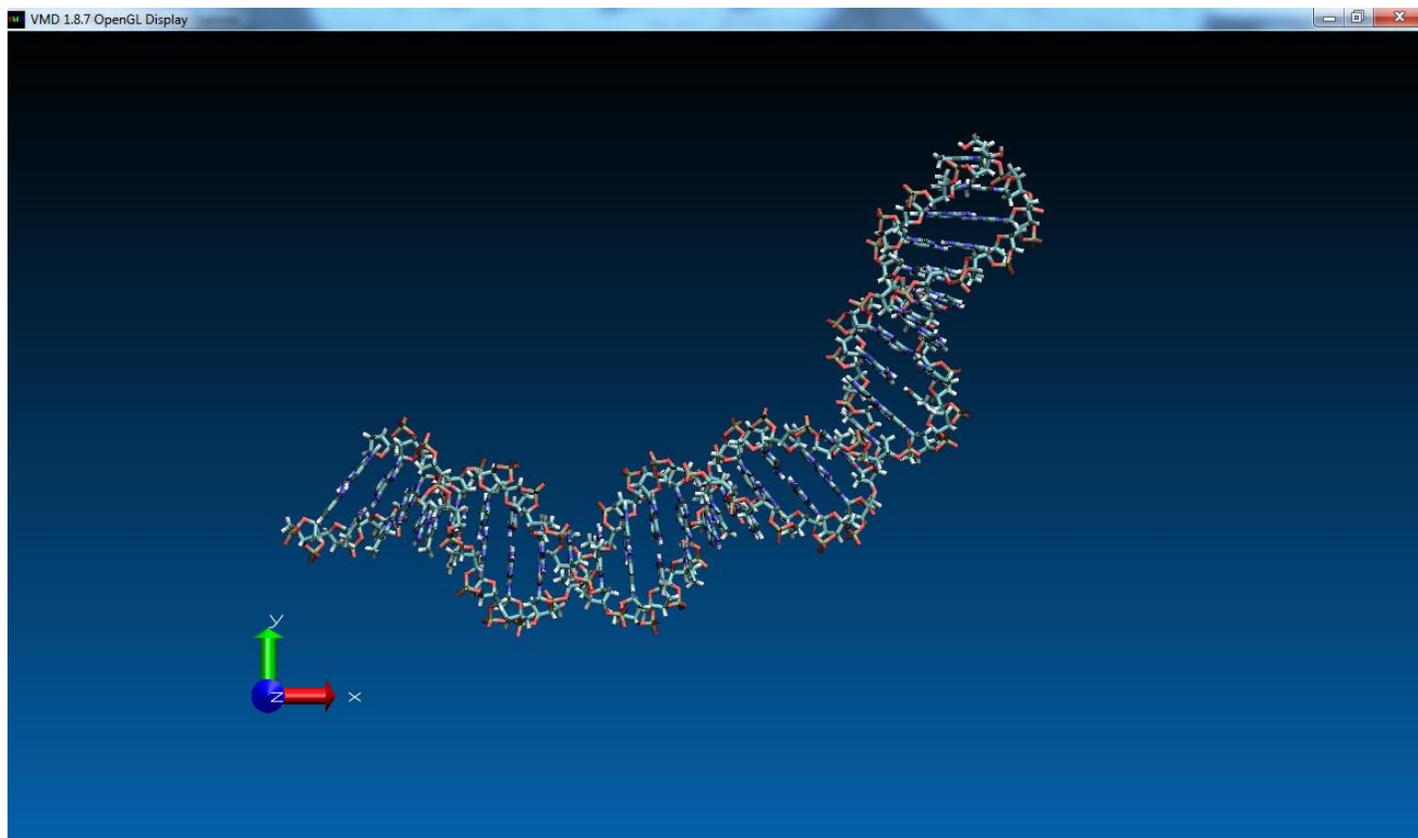


Los **Servicios Básicos** proporcionan acceso a los recursos fundamentales de Windows, entre estos se incluyen los archivos de sistema, dispositivos, procesos, hilos, registros o manejo de errores. El **GDI** (Graphics Device Interface – Interfaz de Dispositivo Gráfico) es una interfaz que permite trabajar con gráficos y permite la interacción entre dispositivos gráficos como, el monitor, la impresora o incluso archivos. La **Interfaz de Usuario** proporciona funcionalidad para crear ventanas y distintos tipos de controles como botones o menús. Los **Servicios de Red** proporcionan acceso a las capacidades de redes de Windows.

Los lenguajes de programación C y C++ pueden usar WinAPI de forma directa, es decir, las bibliotecas de esta API fueron creadas en C y C++, por lo que los demás lenguajes de programación llaman a WinAPI de forma indirecta.

2.5 Visual Molecular Dynamics

La aplicación VMD (Visual Molecular Dynamics – Dinámica Molecular Visual) es un ejemplo de ambiente virtual que permitirá ejemplificar el objetivo de esta tesis. Permite navegar a través de modelos moleculares virtuales, así como manipularlos y modificar sus propiedades visuales, entre otras tareas.^[13] Por lo que cumple con las características más importantes e indispensables dentro de un sistema de navegación virtual. VMD cuenta con varias ventanas de trabajo. A continuación se muestra la ventana de navegación:

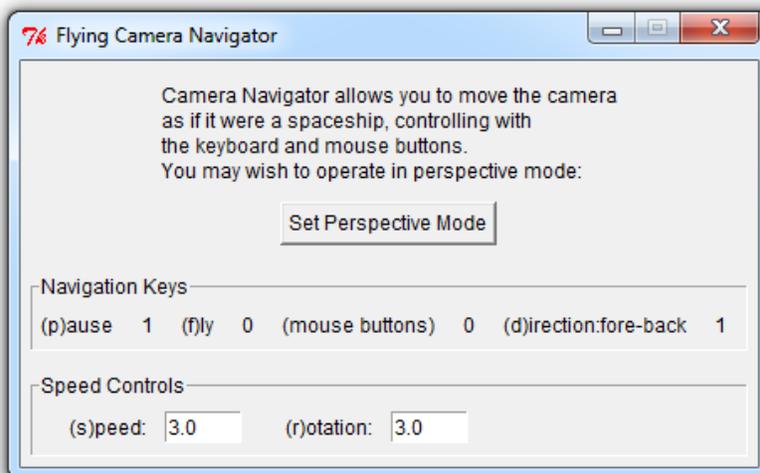


La siguiente ventana de VMD muestran las opciones de teclas a utilizar para navegar en este ambiente virtual:



Esta aplicación cuenta con dos modos de navegación virtual. El primero se realiza con las teclas mencionadas en la ventana anterior, es decir, el teclado permite manipular la cámara desde donde se ve el objeto en la ventana de navegación. El segundo se realiza con el ratón, ya que es posible rotar el modelo molecular al mantener presionado el botón izquierdo del ratón y moverlo en el sentido deseado de rotación.

Sólo tres teclas intervienen en la manipulación del modelo virtual para configurar el modo del ratón, de manipulación del modelo a navegación, así como pausar el movimiento de navegación o cambiar de dirección:



En resumen, VMD es una aplicación que depende de dos dispositivos periféricos, el ratón y el teclado, para su completa manipulación.