

DISEÑO Y CONSTRUCCIÓN DE UNA INTERFAZ HAPTICA DE SEIS GRADOS DE LIBERTAD



**YAMID ESTIVEN OTAYA
MARCO FIDEL PIAMBA**

**Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Electrónica, Instrumentación y Control
Línea de I+D en Robótica y Control
Ingeniería en Automática Industrial**

Popayán, Junio 2014

DISEÑO Y CONSTRUCCIÓN DE UNA INTERFAZ HAPTICA DE SEIS GRADOS DE LIBERTAD

**YAMID ESTIVEN OTAYA
MARCO FIDEL PIAMBA**

**Tesis presentada a la Facultad de Ingeniería
Electrónica y Telecomunicaciones de la
Universidad del Cauca para la obtención del
Título de**

Ingeniero en Automática Industrial

**Director:
Phd. Oscar Andrés Vivas**

Popayán, Junio 2014

Hoja de Aprobación

Director _____
Phd. Oscar Andrés Vivas

Jurado _____

Jurado _____

Fecha de sustentación: Popayán, de 2014

DEDICATORIA

A Dios por ser mi guía y mi amparo durante los momentos difíciles de mi vida, por darme la fortaleza necesaria para seguir adelante a pesar de los obstáculos. Por bendecirme cada día con el don de la vida y permitirme culminar esta etapa.

A mi madre por haberme dado su amor incondicional, por sus consejos y ejemplo de vida, porque me enseñó que el obrar correctamente tiene su recompensa y que lo más importante es disfrutar cada momento que te regala la vida. Agradezco a la vida por haberme regalado en vida a la mejor mamá del mundo y que ahora desde el cielo es el ángel que me protege.

A ti mamá te dedico con especial y profundo amor este triunfo que más que sentirlo mío es completamente tuyo.

A mi padre y mis hermanos por haber estado junto a mí brindándome su apoyo, amor y cariño; por haber fomentado en mí el deseo de superación y el anhelo de lograr éxitos en la vida.

A mis familiares por ofrecerme su apoyo, cariño e impulsarme a lograr cada uno de mis sueños.

A mi novia por darme su amor y comprensión; porque gracias a ella las tristezas y dificultades son más fáciles de sobrellevar.

A mis amigos y profesores por todo el tiempo compartido, sus enseñanzas, su paciencia, sus consejos pero sobre todo su cariño y su amistad sincera.

Yamid Estiven Otaya Bravo

AGRADECIMIENTOS

A Dios por habernos acompañado y guiado a lo largo de nuestra carrera, por habernos dado la sabiduría y entendimiento necesarios para enfrentar cada reto con tenacidad y esmero, por ser nuestra fortaleza en los momentos de debilidad; por protegernos y brindarnos una vida llena de aprendizajes y experiencias.

Agradecemos a nuestro director de tesis, ingeniero Andrés Vivas, quien con sus conocimientos, su experiencia y su motivación logro que culmináramos este proyecto.

Agradecemos de manera especial y sincera a los ingenieros Carlos Felipe Rengifo y German Vaca, por sus importantes aportes en el desarrollo de este proyecto, por su disponibilidad y paciencia. Porque su participación ha enriquecido el trabajo realizado.

A nuestros amigos y colegas Yamir Hernando Bolaños, Tito Manuel Piamba Y Fernanda Pineda quienes siempre nos apoyaron y demostraron su amistad sincera.

RESUMEN

El trabajo que se desarrolla a continuación busca enlazar al usuario con un entorno virtual generando un contacto físico por medio del sentido del tacto. Para generar este tipo de ambientes se requiere utilizar un dispositivo hardware, el cual puede ser manipulado por el usuario y un entorno tridimensional virtual que permita simular un escenario de trabajo.

Este prototipo está basado en interfaces hápticas ya creadas como: HAPTIC 3R, FALCON, VIRTUOSE, PHANTOM PREMIEUM 3.0. La primera interfaz mencionada es un proyecto que se desarrolló en la Universidad del Cauca, que permitió tener un contacto más cercano con este tipo de interfaces hápticas; las demás interfaces se encuentran actualmente en el mercado.

HAPTIC 6DOF utiliza un modelo geométrico directo ya establecido con el cual se posiciona un objeto móvil en un punto deseado dentro de un entorno 3D, una vez se conocen los ángulos que genera cada articulación.

Para la detección de colisiones se implementa un entorno 3D que permite realimentar las fuerzas por medio de la interfaz HAPTIC 6DOF; conjuntamente se implementa una interfaz de usuario que permite habilitar la comunicación entre el entorno 3D y Arduino, adicionalmente esta interfaz permite seleccionar los tres tipos de realimentación de fuerzas que se implementaron.

Para este proyecto se realizó inicialmente la simulación de HAPTIC 6 DOF mediante la herramienta matemática MATLAB®. Consecutivamente se realizó el diseño de las piezas y su estructura mediante el software CAD SolidEdge®, con el fin de tener una idea general de la interfaz.

Una vez se realizaron las anteriores tareas, se procedió a seleccionar la instrumentación de la interfaz. Gracias a los torques obtenidos mediante la simulación, la disponibilidad en el mercado nacional y su costo; se logró determinar los dos tipos de actuadores que se van a utilizar. Para el proceso de censado se utilizaron potenciómetros logarítmicos, que de acuerdo al tipo de articulación se han denominado como sensores secundarios (articulaciones de posicionamiento) y sensores de orientación (articulaciones de orientación).

Posteriormente se enviaron los diseños CAD de las piezas, para realizar los cortes en las placas de acrílico, gracias a una maquina láser.

Teniendo los componentes electrónicos, instrumentación, las piezas terminadas, y otros elementos mecánicos se inicia con la construcción de las tres primeras articulaciones. Consecutivamente se realizan las primeras pruebas de adquisición de datos.

Paralelamente a estas tareas se inicia con la instalación del software que permite realizar la comunicación entre el computador y la interfaz; con el fin de lograr un realismo visual apropiado.

Una vez realizadas las primeras pruebas de comunicación serial con resultados positivos, se continuo con el acople de las siguientes 3 articulaciones de posicionamiento. Luego se realizaron las respectivas pruebas entre el objeto móvil (esfera) dentro del objeto fijo (cubo).

Posteriormente se realizaron las pruebas de realimentación de fuerzas una vez se presente el evento de colisión. La siguiente tarea fue el montaje total de las articulaciones, con las cuales se realizaron las últimas correcciones del montaje y pruebas de funcionamiento.

ABSTRACT

The work being done then seeks to link the user with a virtual environment generating physical contact through the sense of touch. To generate this type of environment are required to use a hardware device, which the user can be manipulated and three-dimensional virtual environment to simulate a working scenario.

This prototype is based on haptic interfaces already created as HAPTIC 3R, FALCON, VIRTUOSE, PHANTOM PREMIEUM 3.0. The first interface is referred to a project that was developed at the University of Cauca, which allowed to have closer contact with this kind of haptic interfaces; other interfaces are on the market.

HAPTIC 6DOF uses a direct geometric model already established which positioned a mobile object at a desired point within a 3D environment, when is known the angle that generating each articulations known.

For the Collision detection we implement 3D environment that allows feedback forces through HAPTIC 6DOF interface. Together a user interface that allows you to enable communication between the 3D environment and Arduino, additionally this interface allows you to select three types of force feedback that were implemented.

For this project first we perform a simulation the de HAPTIC 6 DOF through using MATLAB® mathematical tool. Consecutively the design of parts and their structure by® CAD SolidEdge software, in order to have a general idea of the interface.

Then we proceeded to select the instrumentation of the interface. Thanks to the torques obtained by simulation, the availability in the domestic market and its cost; it was determined the two types of actuators that are to be used. Regarding the process of census, we used logarithmic potentiometers for joint orientation and the position.

Subsequently CAD designs of the pieces were sent to cuts in acrylic plates, thanks to a laser machine.

Considering the electronics components, instrumentation, finished parts, and other mechanical elements begins with the construction of the first three joints. Consecutively the first tests are performed data acquisition.

Parallel to these activities begins with the installation of software that allows communication between the computer and the interface; in order to achieve an appropriate visual realism.

After making the first tests of serial communication with positive results, we continue with the coupling constant of the following 3 joints positioning. Then were performed the respective tests between the mobile object (sphere) within the fixed object (cube)

Later tests were performed the test force feedback once the collision event occurs. The next task was the total assembly of the joints, which we do the the latest fixes of mount and performance test.

Tabla de contenido

1	INTRODUCCIÓN.....	16
2	SISTEMAS HAPTICOS	18
2.1	INTRODUCCIÓN.....	18
2.2	Elementos de una interfaz háptica.....	19
2.2.1	Usuario.....	19
2.2.2	Interfaz háptica.....	19
2.3	Escenario virtual.....	19
2.4	CLASIFICACIÓN DE LAS INTERFACES HÁPTICAS.....	20
2.4.1	Dispositivos de impedancia.....	20
2.4.2	Dispositivos de admitancia.....	20
2.5	INTERFACES HÁPTICAS EXISTENTES	21
2.5.1	Interfaz Phantom.....	21
2.5.2	Dispositivo esférico háptico 3-DOF.....	21
2.5.3	Interfaz VIRTUOSE 3D15-25.....	22
2.5.4	Interfaz Novint Falcon.....	22
2.5.5	Inmersión Cyberforce.....	23
2.5.6	The Freedom 7s.....	23
2.5.7	Interfaz Haptic 3R.....	24
2.6	PROYECTOS QUE SE ENCUENTRAN EN CURSO.....	25
2.6.1	Magnetic Levitation Haptic Consortium.....	25
2.6.2	Proyecto LHifAM.....	25
3	MODELADO, ESTRUCTURA Y SIMULACIÓN.....	27
3.1	MODELADO HAPTIC 6 DOF.....	27
3.1.1	Modelo matemático HAPTIC DOF.....	27
3.1.2	Modelo geométrico directo.....	28
3.1.3	Modelo geométrico inverso.....	30
3.1.4	Modelo dinámico.....	31
3.2	SIMULACIÓN HAPTIC 6DOF.....	34
3.3	DISEÑO DE UN CONTROLADOR POR PAR CALCULADO (CTC)	35
3.4	SINTONIZACIÓN DEL CONTROLADOR CTC.....	36

3.5	SEGUIMIENTO DE TRAYECTORIAS	36
4	DISEÑO MECÁNICO Y CONSTRUCCIÓN DE HAPTIC 6DOF	39
4.1	Diseño CAD de HAPTIC 6DOF.....	39
4.2	Selección de materiales y componentes electrónicos para la construcción de HAPTIC 6DOF.	43
4.2.1	Material para la construcción de las piezas.	43
4.2.2	Selección de actuadores.	44
4.2.3	Selección de sensor.	47
4.2.4	Hardware adicional.....	47
4.2.5	Hardware de comunicación.	52
4.2.6	Tarjeta de adquisición Arduino	53
4.3	Articulaciones HAPTIC 6DOF	53
4.3.1	Articulaciones de posicionamiento.	54
4.3.2	Articulaciones de orientación.....	57
4.4	CONSTRUCCIÓN DEL SISTEMA DE REALIMENTACIÓN.....	59
5	HERRRAMIENTAS SOFTWARE.....	66
5.1	SOFTWARE DE CONTROL	66
5.1.1	Herkulex manager	66
5.1.2	Plataforma Arduino.....	67
5.2	SOFTWARE DE INTERFAZ DE USUARIO	67
5.2.1	Blender.....	68
5.2.2	Visual Studio	68
5.2.3	QT	68
5.2.4	LIBRERIAS VTK	69
5.2.5	Librerias VCOLLIDE.....	69
5.2.6	CMAKE	69
5.3	CONSTRUCCIÓN DE LA INTERFAZ HÁPTICA.....	69
5.3.1	Comunicación interfaz – arduino	76
5.3.2	Detección de colisiones.....	78
5.3.3	Programación Arduino.....	78
6	RESULTADOS.	80
6.1	resultados EN LA zona libre.....	80
6.2	resultados zona de REALIMENTACIÓN.	83

6.3	RESULTADOS PARA ORIENTACIÓN	85
7	CONCLUSIONES Y TRABAJOS FUTUROS	89
7.1	trabajos futuros	90
8	Bibliografía	93

Lista de Tablas

Tabla 3. 1 Parámetros Geométricos HAPTIC 6 DOF.	28
Tabla 3. 2 Masas respectivas de las articulaciones.	33
Tabla 3. 3 Distancia a centros de masa de cada articulación en ejes x, y, z.	33
Tabla 3. 4 Primer momento de inercia.	33
Tabla 3. 5 Segundo momento de inercia.	34
Tabla 3. 6 Ganancias del controlador	36
Tabla 4. 1 Datos obtenidos en laboratorio.	49
Tabla 4. 2 Relación entre ángulo v/s elongación del resorte.	63
Tabla 4. 3 Relación entre ángulo v/s lecturas potenciómetro.	63
Tabla 4. 4 Relación entre datos actuador principal-sensor secundario-resorte.	64
Tabla 5. 1 Explicación de líneas de código.	74
Tabla 5. 2 Explicación de líneas de código	75
Tabla 5. 3 Configuración Tserial (comunicación serial) [6].	76
Tabla 5. 4 Espacios de trabajo interfaz – entorno 3D.....	77
Tabla 5. 5 Comandos de los servomotores Herkulex en ARDUINO.....	79

Lista de Figuras

Figura 2. 1 Elementos de una interfaz háptica.	19
Figura 2. 2 PHANTOM Omni (a), PHANTOM Premium 6DOF (b).....	21
Figura 2. 3 Dispositivo esférico háptico 3-DOF.	22
Figura 2. 4 Haptic Device VIRTUOSE 3D15-25.	22
Figura 2. 5 Dispositivo Háptico Novint Falcon.	23
Figura 2. 6 Immersion Cyberforce.....	23
Figura 2. 7 The Freedom 7S [20].	24
Figura 2. 8 Interfaz Haptic 3R.	24
Figura 2. 9 The Maglev Haptic.	25
Figura 2. 10 Robot LHfAM (interfaz háptica para la mantenibilidad aeronáutica). 26	
Figura 3. 1 Estructura cinemática HAPTIC 6DOF.	27
Figura 3. 2 Propiedades físicas del cuerpo de rotación θ_1	32
Figura 3. 3 Esquema Haptic 6DOF.	35
Figura 3. 4 Esquema del control CTC.	36
Figura 3. 5 Trayectoria circular.	37
Figura 3. 6 Grafica del error cartesiano.....	37
Figura 3. 7 Torque obtenido para cada articulación.	38
Figura 4. 1 Diseño de HAPTIC 6DOF.	39
Figura 4. 2 Actuador principal No 1 anclado a base.	40
Figura 4. 3 Cuerpo de rotación θ_1	40
Figura 4. 4 Cuerpo de rotación θ_2	41
Figura 4. 5 Cuerpo de rotación θ_3	41
Figura 4. 6 Cuerpo de rotación θ_4	42
Figura 4. 7 Cuerpo de rotación θ_5	42
Figura 4. 8 Cuerpo de rotación θ_6	43
Figura 4. 9 Servo digital: HERKULEX DRS 0101.....	45
Figura 4. 10 Conexión en serie (a) y distribución de pines (b)	46
Figura 4. 11 Servo análogo H-KING 15178.	46
Figura 4. 12 Potenciómetro logarítmico.	47
Figura 4. 13 Placas ancladas.....	48
Figura 4. 14 Placas desancladas.	48
Figura 4. 15 Características generales del Resorte.	49
Figura 4. 16 Mapeo de datos obtenidos en laboratorio.	50
Figura 4. 17 Linealización de los datos.	50
Figura 4. 18 Regulador de voltaje.	51
Figura 4. 19 Fuente de alimentación.....	52
Figura 4. 20 Tarjeta de expansión.	52
Figura 4. 21 Tarjeta ATmega2560.	53
Figura 4. 22 Vista inferior del actuador (a) y vista frontal de la articulación (b).....	54
Figura 4. 23 Soporte de la segunda articulación.	54
Figura 4. 24 Placas de segunda articulación (a) y soporte de acrílico para tercer actuador (b).	55
Figura 4. 25 Placas (a) y rieles del tercer actuador (b).....	55

Figura 4. 26 Componentes de las articulaciones de posicionamiento.	56
Figura 4. 27 Dirección de las fuerzas que actúan en ausencia de colisiones.	56
Figura 4. 28 Dirección de las fuerzas que actúan con colisión.	57
Figura 4. 29 Articulaciones de posicionamiento.	57
Figura 4. 30 Piezas cuarta articulación.	58
Figura 4. 31 Piezas quinta articulación.	58
Figura 4. 32 Piezas sexta articulación.	59
Figura 4. 33 Articulaciones de orientación.	59
Figura 4. 34 Diagrama primer evento.	61
Figura 4. 35 Diagrama segundo evento.	61
Figura 4. 36 Tipos de realimentación.	62
Figura 5. 1 Software Herkulex Manager.	67
Figura 5. 2 Diagrama de flujo del funcionamiento.	70
Figura 5. 3 Desarrollo de una esfera en Blender.	71
Figura 5. 4 Integración de los objetos 3D.	71
Figura 5. 5 Desarrollo de la Interfaz en Qt.	72
Figura 5. 6 Interfaz de usuario HAPTIC 6DOF.	75
Figura 6. 1 Dos puntos de la línea en el entorno 3D – eje x.	81
Figura 6. 2 Dos puntos de la línea en HAPTIC 6DOF – eje x.	81
Figura 6. 3 Dos puntos de la línea en el entorno 3D – eje Y.	82
Figura 6. 4 Dos puntos de la línea en HAPTIC 6DOF – eje y.	82
Figura 6. 5 Dos puntos de la línea del entorno 3D – eje z.	83
Figura 6. 6 Dos puntos de la línea en HAPTIC 6DOF – eje y.	83
Figura 6. 7 Realimentación suave.	84
Figura 6. 8 Realimentación media.	84
Figura 6. 9 Realimentación fuerte.	85
Figura 6. 10 Puntos de rotación en el eje X – entorno 3D.	85
Figura 6. 11 Puntos de orientación en el eje x – HAPTIC 6DOF.	86
Figura 6. 12 Puntos de rotación en el eje Y.	86
Figura 6. 13 Puntos de orientación en el eje y – HAPTIC 6DOF.	87
Figura 6. 14 Puntos de rotación en el eje Z.	87
Figura 6. 15 Puntos de orientación en el eje z – HAPTIC 6DOF.	88
Figura 7. 1 Diagrama de bloques del sistema propuesto.	90

1 INTRODUCCIÓN

El termino háptica significa principalmente, todo aquello referido al contacto. El término háptico proviene de un término griego que significa “capaz de coger o asir”. La morfología del término está en las raíces hap-tic ('hap-tik) del griego *haptiko*, adjetivo asociado al sentido del tacto; en el vocabulario científico se designa *haptic interface* al dispositivo que utiliza al tacto como sentido esencial para su funcionamiento. Por tanto la función de un sistema háptico es generar la sensación de tocar, manipular y sentir objetos desarrollados en entornos tridimensionales por medio de un dispositivo físico [1].

Los primeros estudios sobre interfaces hápticas se realizaron en los años 90, volviéndose un área de investigación multidisciplinar, debido a que reúne distintos campos de la ingeniería, principalmente áreas como la mecatrónica, informática y el control digital. Este tipo de interfaces buscaban generar estímulos a través de los sentidos.

Los dos sentidos más utilizados por el ser humano son la audición y la visión ya que son los principales medios de interacción con el entorno. Luego se encuentran los sentidos del gusto y el olfato; posteriormente se encuentra el sentido del tacto el cual permite tener una información bidireccional, debido a que permite crear y sentir estímulos táctiles; con el propósito de generar una sensación de inmersión, con posibilidad de interacción por medio de los entornos virtuales [2].

En los últimos años se han realizado distintas aplicaciones en diferentes campos como: medicina [3], videojuegos [4], diseño mecánico [5], aunque las aplicaciones más relevantes se encuentran en el campo de la telemedicina y la rehabilitación de pacientes que presentan algún tipo de discapacidad física [6].

Este proyecto propone la creación de una interfaz háptica de 6 grados de libertad basada en un primer prototipo de 4 grados de libertad, desarrollado en la Universidad del Cauca hace 3 años llamado: Haptic 3R (sistema de interfaz háptica para el control de posición en un espacio tridimensional virtual) [7]. Además este prototipo se basa en las interfaces existentes actualmente en el mercado.

El proyecto busca modelar un prototipo para mirar su dinámica y poder simular el comportamiento frente a una trayectoria previamente definida. Además se creará una interfaz tridimensional para comprobar el adecuado comportamiento del prototipo el cual se llamará HAPTIC 6 GDL por poseer 6 grados de libertad; 3 para posicionar la interfaz en un punto deseado y 3 para generar rotación en los 3 ejes.

Por tanto el propósito de este proyecto es disminuir los costos que implica la construcción de una interfaz háptica con estas características y adicionalmente generar nuevos conocimientos relacionados con la implementación de interfaces hápticas que sean fácilmente reproducibles.

Este proyecto contiene cinco capítulos que se relacionan de la siguiente manera: el capítulo uno contiene la información general sobre interfaces hápticas, campos de aplicación y proyectos desarrollados; en el capítulo dos se encuentra explicado lo relacionado con el modelado matemático, estructura y simulación. El tercer capítulo muestra el diseño y construcción de HAPTIC 6DOF, materiales, actuadores y sensores. El diseño mecánico, la construcción tanto del sistema de realimentación como de las articulaciones de posicionamiento y de orientación están explicados en el capítulo 4.

Dentro del capítulo 5 se exponen los programas que se utilizaron para el software de control y la realización de la interfaz de usuario, concluyendo con la construcción total del entorno software sobre el cual se va a trabajar.

En el capítulo 6 se explican los resultados obtenidos tanto para la zona libre como para la zona de realimentación. Finalmente en el capítulo 7 se consignan las conclusiones y trabajos futuros.

2 SISTEMAS HAPTICOS

2.1 INTRODUCCIÓN

Las interfaces hápticas permiten al usuario interactuar, ya sea enviando información a un simulador o recibiendo información desde el entorno de simulación. En el caso de las interfaces hápticas este proceso se realiza mediante un dispositivo de entrada/ salida que permite que el usuario tenga un contacto físico con el computador; las interfaces hápticas pueden generar dos tipos de realimentación, una realimentación de fuerza y otra táctil. El primer tipo de realimentación aplica fuerzas sobre el usuario; y el segundo permite detallar la superficie con la que se está interactuando. Aunque algunas interfaces tienen la capacidad de generar los dos tipos de realimentación [8].

Por tanto, se puede decir que las interfaces hápticas son dispositivos bidireccionales que permiten al usuario simular sensaciones de fuerza o tacto. Existen dispositivos hápticos y teleoperados que buscan aumentar la sensación de inmersión en el espacio de trabajo; dependiendo del sistema que se utilice la zona remota puede o no existir.

En los sistemas teleoperados la zona remota posee un robot para poder manipular. A diferencia de los sistemas hápticos en donde la zona remota no existe físicamente, en cambio esta es simulada por medio de un software. Por ende la única diferencia entre estos dos sistemas está en la señal de realimentación [2].

Según [7], los factores principales para obtener un buen comportamiento del sistema son: El control del dispositivo háptico, la naturaleza de la señal realimentada que proviene de un entorno virtual o físico y las características mecánicas del dispositivo.

2.2 ELEMENTOS DE UNA INTERFAZ HÁPTICA

En todo sistema háptico actúan tres componentes esenciales como se muestra en la Figura 2.1

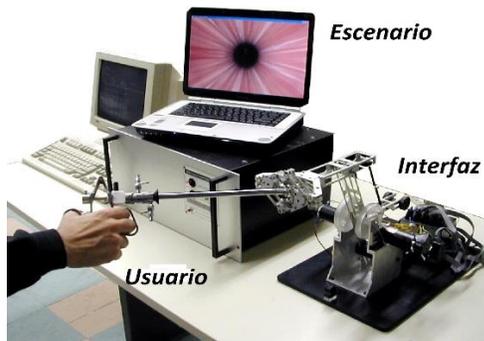


Figura 2. 1 Elementos de una interfaz háptica.

2.2.1 Usuario.

Forma parte esencial ya que determina el comportamiento y la estabilidad del sistema por hacer parte del lazo de control.

2.2.2. Interfaz háptica.

Es un dispositivo que permite enviar información por medio de los sensores para conocer la posición en la que se encuentra el usuario y recibir información a través de los actuadores, restituyendo la fuerza de contacto con el escenario tridimensional. Además ayuda a determinar el espacio de trabajo de la interfaz.

2.2.1 Escenario virtual.

Una de las tareas principales del escenario virtual es recolectar información de la localización en el espacio de los objetos que la componen y además guardar características gráficas de cada objeto a través de una base de datos.

Otra tarea del escenario virtual es calcular las zonas de contacto entre el objeto que manipula el usuario y el escenario virtual. Uno de los componentes esenciales en el escenario es el software de control que permite calcular y dirigir las fuerzas que se deben enviar al usuario de acuerdo a la información geométrica de contacto que se obtiene gracias al módulo de detección de colisiones [1].

2.3 CLASIFICACIÓN DE LAS INTERFACES HÁPTICAS

Una clasificación que se puede adoptar está basada en el tipo de control que requiere un dispositivo háptico como lo explica [8].

2.3.1 Dispositivos de impedancia.

Este reacciona con una fuerza de acuerdo a la posición del dispositivo determinada por el usuario. El paradigma es “el desplazamiento entra – la fuerza sale”. Este tipo de dispositivos incluyen las siguientes características:

- El movimiento es captado por medio de los sensores de desplazamiento.
- El desplazamiento es transmitido por medio del controlador hardware hacia al simulador software.
- El simulador identifica si hay o no colisión y su magnitud.
- El controlador hardware recibe la señal.
- El controlador hardware ordena al servo motor mantener una determinada posición.
- El servo motor realiza la fuerza necesaria para mantener la posición.

2.3.2 Dispositivos de admitancia.

Este dispositivo reacciona con el movimiento o trayectoria que se le aplica gracias a la fuerza que ejerce el usuario. Lo que hace esta interfaz es responder ante cambios en la aceleración, la velocidad y la posición. El paradigma que utiliza es “fuerza entra – desplazamiento sale”.

Otro tipo de clasificación según [1], se debe al tipo de actuador que posea:

- Dispositivos con actuadores hidráulicos.
- Dispositivos con actuadores neumáticos.
- Dispositivos con actuadores eléctricos.

Y dependiendo del anclaje:

- Dispositivos fijos.
- Dispositivos portátiles.

2.4 INTERFACES HÁPTICAS EXISTENTES

2.4.1 Interfaz Phantom.

Son dispositivos hápticos de alta precisión que permiten al usuario una mayor sensación al manipular objetos virtuales. Este tipo de interfaces ofrecen mayores áreas de trabajo, soportan mayores fuerzas y generan menores fricciones en el movimiento; están diseñadas para el campo comercial e investigativo y son distribuidas por la empresa Sensable Technologies.

En la Figura 2. 2. (a) Se tiene la interfaz PHANTOM Omni de fácil uso y rápida instalación; la cual posee un diseño portátil y es de tamaño compacto. En la Figura 2. 2. (b) se muestra otro modelo: PHANTOM Premium [9].

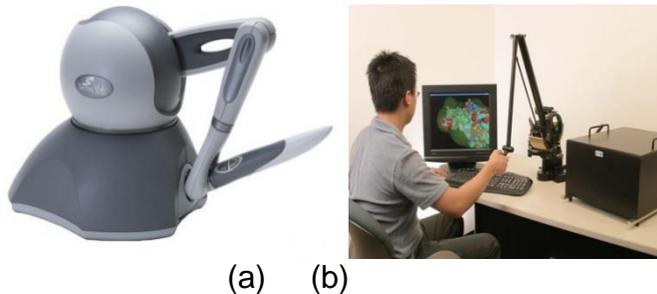


Figura 2. 2 PHANTOM Omni (a), PHANTOM Premium 6DOF (b).

2.4.2 Dispositivo esférico háptico 3-DOF.

Este dispositivo es también llamado SHADE por sus siglas en inglés: *Spherical Haptic Device*. SHADE posee 3 grados de libertad lo que permite una simplificación en el sistema; además su diseño es más ergonómico y su geometría esférica admite una rotación alrededor de un punto fijo donde se encuentra ubicada la mano del usuario, como se muestra en la Figura 2. 3. Esta interfaz utiliza como algoritmo de control de potencia un PID clásico mejorado con compensación feed-forward con el fin de mejorar el rendimiento [10].

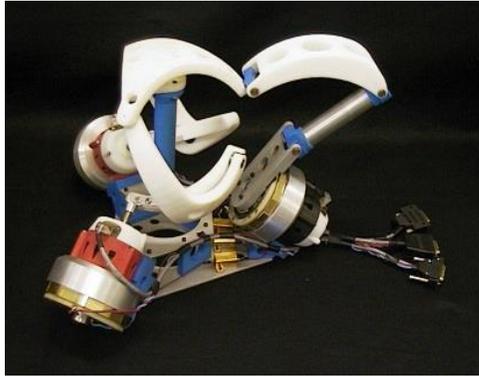


Figura 2. 3 Dispositivo esférico háptico 3-DOF.

2.4.3 Interfaz VIRTUOSE 3D15-25.

Esta interfaz posee 3 grados de libertad y cuenta con un sensor de proximidad que hace que los motores se activen cuando el usuario entra en contacto con la interfaz y se desactivan cuando se ha dejado de utilizar la interfaz.

La interfaz VIRTUOSE 3D15-25 que se muestra en la Figura 2. 4. Es comercializada por la empresa HAPTION [11].



Figura 2. 4 Haptic Device VIRTUOSE 3D15-25.

2.4.4 Interfaz Novint Falcon.

Este dispositivo fabricado por Novint Technologies Inc. que se muestra en la Figura 2. 5, posee una arquitectura paralela con un amplio espacio de trabajo y es capaz de soportar fuerzas de trabajo altas, fue diseñada inicialmente para aplicaciones de entretenimiento en escenarios 3D reemplazando a dispositivos periféricos como el joystick y el ratón [12].



Figura 2. 5 Dispositivo Háptico Novint Falcon.

2.4.5 Inmersión Cyberforce.

Esta interfaz de retroalimentación de fuerza combina 2 productos: el CyberGrasp y la CyberGlobe, que unidos proporcionan un fuerza de respuesta para la mano y los dedos; el principal propósito de esta asociación de interfaces es la generación de fuerzas individuales, que permitan explorar intuitivamente e interactuar con los objetos gráficos que se encuentran en los entornos 3D. La CyberForce se muestra en la Figura 2. 6, y se fabrica bajo licencia de SensAble Technologies, Inc. [13].



Figura 2. 6 Immersion Cyberforce.

2.4.6 The Freedom 7s.

Fabricada por la empresa MPB technologies Inc. Esta interfaz háptica fue construida sobre la interfaz Freedom 6s y posee alta fidelidad de realimentación de fuerza. Se utiliza en entornos donde se deba manipular un mango similar a unas tijeras como se observa en la Figura 2. 7, se usa en campos como la cirugía robótica, tutoría médica, manejo de materiales sensibles en entornos de salas limpias etc. [14].

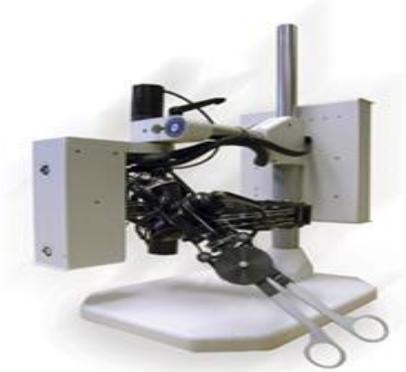


Figura 2. 7 The Freedom 7S [20].

2.4.7 Interfaz Haptic 3R.

Este dispositivo fue creado en la Universidad del Cauca en el año 2011. Como se observa en la Figura 2.8, su estructura posee cuatro grados de libertad. Los tres primeros permiten posicionar un objeto tridimensional. El último grado de libertad proporciona rotación al objeto que se encuentra en un determinado espacio de trabajo. Este prototipo fue construido para generar realimentación de fuerzas; ya sea ofreciendo un esfuerzo mecánico o deteniéndose en el momento que se presente una colisión en el entorno tridimensional.

Para su visualización se creó una interfaz de usuario utilizando Visual Estudio Express 2008 y una interfaz 3D que utiliza OpenGL vinculándolo a C#. En la Figura 2. 2 se muestra la interfaz Haptic 3R [7].



Figura 2. 8 Interfaz Haptic 3R.

2.5 PROYECTOS QUE SE ENCUENTRAN EN CURSO

2.5.1 Magnetic Levitation Haptic Consortium.

Este consorcio crea dispositivos de 6 grados de libertad; eliminando los componentes mecánicos y utilizando una pieza llamada “flotor”, que levita gracias a campos magnéticos generados por estatores. Estos dispositivos poseen sensores ópticos que ayudan a seguir la posición y orientación del “flotor”; logrando una resolución más alta ya que existe una conexión electrodinámica entre la mano y el ordenador [27]. En la Figura 2.9 se observa un par de dispositivos hápticos Maglev para dos manos.



Figura 2. 9 The Maglev Haptic.

2.5.2 Proyecto LHfAM.

El nombre se da por sus siglas en inglés: *Large Haptic Interface for Aeronautic Maintainability*. Este dispositivo junto a un nuevo diseño de muñeca háptica de 3 grados de libertad ha permitido sentir la restitución de fuerza en las tres orientaciones y está siendo utilizado en la reparación del motor de los aviones por la dificultad que se presenta al momento de trabajar entre tantos cables y engranajes. Este dispositivo permite trabajar desde un entorno remoto, simulando las tareas de mantenimiento en los motores de avión gracias a brazos robóticos [15].

En la Figura 2. 10, se puede observar uno de los dispositivos que se desarrollaron para el presente proyecto.

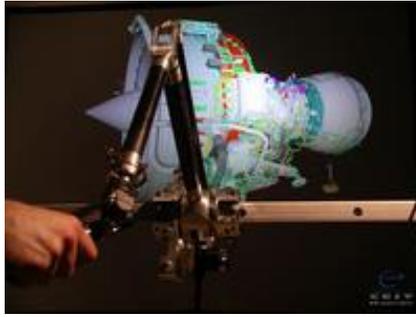


Figura 2. 10 Robot LHI fAM (interfaz háptica para la mantenibilidad aeronáutica).

3 MODELADO, ESTRUCTURA Y SIMULACIÓN

3.1 MODELADO HAPTIC 6 DOF

Esta sección muestra la obtención de los diferentes parámetros para el diseño y modelado del dispositivo háptico HAPTIC 6DOF. De acuerdo a su estructura en serie se diseñarán 3 grados de libertad que proveen la posición del efector final, además de 3 grados de libertad que otorga la orientación del objeto en el espacio cartesiano.

En la Figura 3. 1 se muestra la estructura cinemática del HAPTIC 6DOF, donde todas las articulaciones son de tipo rotoide.

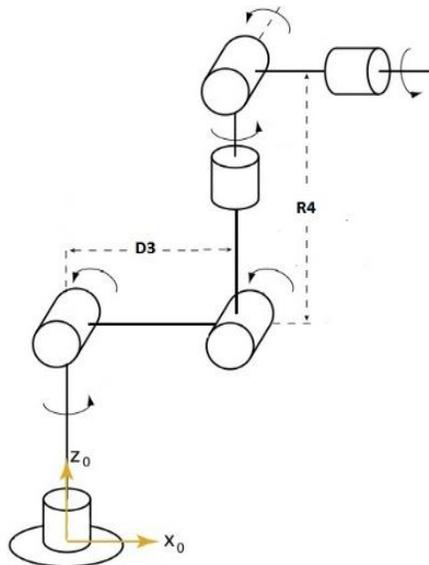


Figura 3. 1 Estructura cinemática HAPTIC 6DOF.

3.1.1 Modelo matemático HAPTIC 6DOF.

En el diseño estructural del HAPTIC 6DOF se generan los cálculos correspondientes para el modelo matemático, el cual permite verificar el funcionamiento de las trayectorias deseadas del sistema. Estos modelos permiten representar geoméricamente el diseño a través de la interfaz.

Como solución para el modelo matemático se utiliza el método Khalil-Dombre [16]. Como resultado de este método se obtiene la siguiente tabla de parámetros geoméricos (Ver Tabla 3. 1)

j	σ_j	α_j	d_j	Θ_j	r_j
1	0	0	0	Θ_1	0
2	0	90	0	Θ_2	0
3	0	0	D3	Θ_3	0
4	0	-90	0	Θ_4	R4
5	0	90	0	Θ_5	0
6	0	-90	0	Θ_6	0

Tabla 3. 1 Parámetros Geométricos HAPTIC 6 DOF.

Donde:

- **j**: Número de la articulación.
- **σ_j** : Tipo de articulación
- **α_j** : Angulo entre los ejes z_{j-1} y z_j correspondiente a una rotación alrededor de x_{j-1} .
- **d_j** : Distancia entre z_{j-1} y z_j a lo largo de X_{j-1} .
- **Θ_j** : Angulo entre los ejes x_{j-1} y x_j correspondiente a una rotación alrededor de z_j .
- **r_j** : Distancia entre x_{j-1} y x_j a lo largo de z_j .

Teniendo la tabla de parámetros geométricos, se puede encontrar el modelo geométrico directo (MGD).

3.1.2 Modelo geométrico directo.

Este modelo permite calcular la posición (x, y, z) y la orientación del órgano terminal en el espacio cartesiano en función de las coordenadas articulares. Para el desarrollo de este modelo es necesario aplicar las matrices de transformación ${}^{j-1}T_j$ de la ecuación 1, a cada articulación, de acuerdo a la Tabla 3. 1 de parámetros obtenida anteriormente.

$${}^{j-1}T_j = \begin{bmatrix} \cos\theta_j & -\text{sen}\theta_j & 0 & d_j \\ \cos\alpha_j \text{sen}\theta_j & \cos\alpha_j \cos\theta_j & -\text{sen}\alpha_j & -r_j \text{sen}\alpha_j \\ \text{sen}\alpha_j \text{sen}\theta_j & \text{sen}\alpha_j \cos\theta_j & \cos\alpha_j & r_j \cos\alpha_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ecuación 1.}$$

De acuerdo a la ecuación 1 obtenemos el conjunto de matrices que al multiplicarse dan como resultado la matriz de transformación ${}^0T_6 = {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6$ que representa el modelo geométrico directo

Para su cálculo se sigue como referencia el ejercicio mostrado en [17].

Finalmente se obtienen los correspondientes vectores de la matriz 0T_6 .

Para abreviar términos tenemos que:

$$C_i = \cos(\theta_j)$$

$$S_i = \text{sen}(\theta_j)$$

$$C_{ij} = \cos(\theta_i + \theta_j)$$

$$S_{ij} = \text{sen}(\theta_i + \theta_j)$$

Obteniendo:

$$s_x = C_1(C_{23}(C_4C_5C_6 - S_4S_6) - S_{23}S_5C_6) - S_1(S_4C_5C_6 + C_4S_6)$$

$$s_y = S_1(C_{23}(C_4C_5C_6 - S_4S_6) - S_{23}S_5C_6) + C_1(S_4C_5C_6 + C_4S_6)$$

$$s_z = S_{23}(C_4C_5C_6 - S_4S_6) + C_{23}S_5C_6$$

$$n_x = C_1(-C_{23}(C_4C_5S_6 - S_4C_6) + S_{23}S_5S_6) + S_1(S_4C_5S_6 + C_4C_6)$$

$$n_y = S_1(-C_{23}(C_4C_5S_6 - S_4C_6) + S_{23}S_5S_6) - C_1(S_4C_5S_6 + C_4C_6)$$

$$n_z = -S_{23}(C_4C_5S_6 + S_4C_6) - C_{23}S_5S_6$$

$$a_x = -C_1(C_{23}C_4S_5 + S_{23}C_5) + S_1S_4S_5$$

$$a_y = -S_1(C_{23}C_4S_5 + S_{23}C_5) - C_1S_4S_5$$

$$a_z = -S_{23}C_4S_5 + C_{23}C_5$$

$$P_x = -C1(S23R4-C2D3)$$

$$P_y = -S1(S23R4-C2D3)$$

$$P_z = C23R4+S2D3$$

En la obtención del MGD del sistema se utilizan las soluciones P_x , P_y y P_z , las cuales permiten conocer la posición cartesiana. La orientación del órgano terminal es determinada por los cosenos directores s , η , α , los cuales nos generan tres vectores unitarios que describen la orientación del referente 6 en el referente 0 y a partir de estos se estiman los ángulos de Euler: Roll, Pitch, Yaw.

ϕ = rotación en z (Roll).

Θ = rotación en y (Pitch).

Ψ = rotación en x (Yaw).

Las formulas con las que se estiman estos ángulos se encuentran explicadas en [16] y se muestran a continuación:

$$\phi = \text{atan2}(S_y, S_x)$$

$$\phi = \text{atan2}(-S_y, -S_x) = \phi + \pi \quad (\text{Ecuación 2})$$

$$\Theta = \text{atan2}(-S_z, C\phi S_x + S\phi S_y) \quad (\text{Ecuación 3})$$

$$\Psi = \text{atan2}(S\phi a_x - C\phi a_y - S\phi \eta_x + C\phi \eta_y) \quad (\text{Ecuación 4})$$

Es necesario aclarar que para un correcto funcionamiento en el entorno 3D el software de recibir los ángulos en el siguiente orden: Ψ , Θ , ϕ .

3.1.3 Modelo geométrico inverso.

El modelo geométrico inverso (MGI) se utilizó para la simulación de la interfaz ya que reconoce todas las posibles soluciones del cálculo de las coordenadas articulares, correspondiente a un situación cartesiana determinada [17]. Para el cálculo de MGI se utiliza el método de Paul (1981), el cual define una ecuación de igualdad entre el MGD y la matriz U_0 deseada (*Ecuación 1*).

$${}^0T_n = {}^0T_1 {}^1T_2 \dots {}^{n-1}T_n$$

$$U_0 = {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6$$

Para encontrar los valores de las variables articulares se utiliza el método de Paul (1981) para las tres primeras ($\theta_1, \theta_2, \theta_3$) y el caso particular de la muñeca para las últimas tres ($\theta_4, \theta_5, \theta_6$) respectivamente [18].

Solución para θ :

$$\theta_1 = \text{atan}(P_y, P_x) \quad \theta_2 = \text{atan}(S2, C2) \quad \theta_3 = \text{atan}(S3, C3)$$

$$\theta'_1 = \theta_1 + 180^\circ$$

La solución θ'_1 demuestra que para una rotación de 180° también puede ser solución del problema, permitiendo alcanzar una posición deseada igual.

La solución para las últimas articulaciones se desarrolla de acuerdo al método descrito en Vivas [17].

$$\theta_4 = \text{atan}(H_z, H_x)$$

$$\theta'_4 = \theta_4 + 180^\circ$$

$$\theta_5 = \text{atan}(S5, C5)$$

$$\theta_6 = \text{atan}(S6, C6)$$

$$\theta'_6 = \theta_6 + 180^\circ$$

$$\text{Donde } H_z = S1a_x - C1a_y \quad \text{y} \quad H_x = C23(C1a_x + S1a_y) + S23a_z .$$

3.1.4 Modelo dinámico.

El modelo dinámico contiene la información necesaria geométrica y dinámica [17] del HAPTIC 6 DOF, de acuerdo a la relación entre las fuerzas aplicadas a los actuadores (motores, sensores, velocidad y aceleraciones de las articulaciones del robot).

Los parámetros dinámicos que hacen parte de las ecuaciones dinámicas de un dispositivo serial son las siguientes:

- Seis términos del tensor de inercia: $XX_j, XY_j, XZ_j, YY_j, YZ_j, ZZ_j$.
- Tres términos del primer momento de inercia: MX_j, MY_j, MZ_j .
- Un término para la masa: M_j .
- Un término para la inercia del motor: I_{aj} .

En el proceso del cálculo de las inercias de la HAPTIC 6DOF se utilizó la herramienta CAD *SolidEdge*®, la cual permite asignar características y generar las inercias de acuerdo al tipo de material que se utiliza. En la siguiente figura se muestra las propiedades físicas que arroja este software.

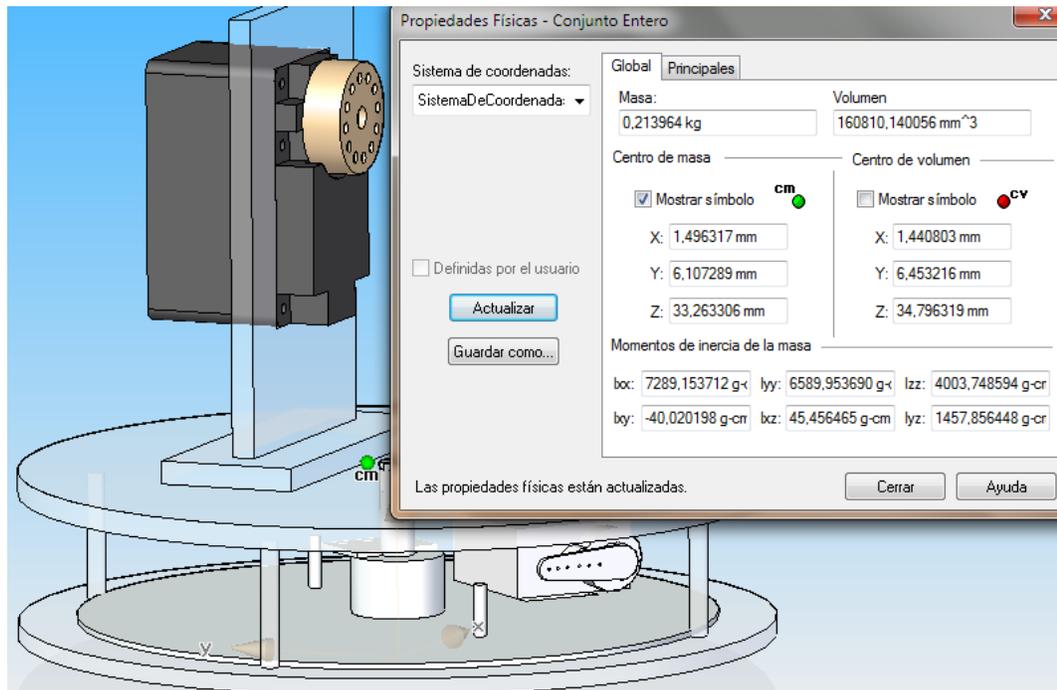


Figura 3. 2 Propiedades físicas del cuerpo de rotación θ_1 .

Para calcular el primer momento de inercia, se multiplica la masa de cada articulación por la distancia al centro de masa de cada una de sus coordenadas en x, y, z para estos cálculos es necesario pasar los todos los valores a metros y kilogramos. Estos datos se obtienen de las propiedades físicas generadas por medio del CAD *SolidEdge*.

A continuación se muestran las tablas de la masa (Tabla 3. 2), distancia a centro de masa (Ver Tabla 3. 3), primer momento de inercia (Tabla 3. 3) y segundo momento de inercia (Ver Tabla 3. 4)

ARTICULACION	MASA DE CADA ARTICULACION(KG)
Articulación 1	M1= 0,213964
Articulación 2	M2= 0,168360
Articulación 3	M3= 0,126535
Articulación 4	M4= 0,069993
Articulación 5	M5= 0,029153
Articulación 6	M6= 0,024761

Tabla 3. 2 Masas respectivas de las articulaciones.

ARTICULACION	Centro de masa en x (m)	Centro de masa en y (m)	Centro de masa en z (m)
Articulación 1	0,001496317	0,006107289	0,033263306
Articulación 2	0,039498452	0,00140084	0,013171842
Articulación 3	0,016761868	0,001128157	0,017489469
Articulación 4	0,000012799	0,010960343	0,051538039
Articulación 5	-0,000036726	-0,000124488	0,042406075
Articulación 6	0	0	-0,012838417

Tabla 3. 3 Distancia a centros de masa de cada articulación en ejes x, y, z.

ARTICULACIÓN	Primer momento de inercia en x	Primer momento de inercia en y	Primer momento de inercia en z
Articulación 1	MX1=0,0003199944	MY1= 0,00130672094	MZ1=0,0071171487212
Articulación 2	MX2= 0,0066485364	MY2= 0,00023583868	MZ2=0,002217604248
Articulación 3	MX3=0,00212095436	MY3=0,00014274413	MZ3=0,002213021229
Articulación 4	MX4=0,00000088887	MY4= 0,00767111397	MZ4=0,0036034929
Articulación 5	MX5= - 0,0000001070	MY5= - 0,0000036291	MZ5=0,0012362643044
Articulación 6	MX6= 0	MY6= 0	MZ6= - 0,00031789162

Tabla 3. 4 Primer momento de inercia.

El segundo momento de inercia o matriz de inercia es arrojado por el software directamente. Está dado en g.cm² por tanto debe hacerse la conversión a Kg.m²

ARTICULACION	Segundo momento de inercia en x,y,z (Kg. m ²)
--------------	---

Articulación 1	XX1=0,00072891537	YY1=0,00065899536	ZZ1=0,00040037485
	XY1=-0,00000400201	XZ1=0,00000454564	YZ1=0,00014578564
Articulación 2	XX2=0,00019688972	YY2= 0,00093765607	ZZ2= 0,00090926643
	XY2= 0,00000135247	XZ2= 0,00009221482	YZ2= 0,00000330605
Articulación 3	XX3= 0,00011681109	YY3= 0,00024414937	ZZ3= 0,00024355506
	XY3= 0,0000034085	XZ3= 0,00004114934	YZ3= 0,000001951
Articulación 4	XX4=0,00026111792	YY4=0,0002294234	ZZ4=0,00005520322
	XY4=0,00000000781	XZ4=0,00000007936	YZ4=0,00005078932
Articulación 5	XX5=0,00006736441	YY5=0,00006739351	ZZ5=0,00000469873
	XY5=0	XZ5=-0,00000007208	YZ5=-0,00000003687
Articulación 6	XX6=0,00000685973	YY6=0,00000685973	ZZ6=0,00000283001
	XY6=0	XZ6=0	YZ6=0

Tabla 3. 5 Segundo momento de inercia.

3.2 SIMULACIÓN HAPTIC 6DOF

La simulación del HAPTIC 6 DOF se desarrolla con la herramienta *MATLAB*®. Gracias a los datos obtenidos anteriormente se realiza la simulación del HAPTIC 6DOF, basándose en los modelos geométricos y dinámicos. En los resultados se busca encontrar un error mínimo de acuerdo al control utilizado.

En la Figura 3. 3, se observa el diseño del sistema en un modelo tridimensional de la interfaz, esto modelo se desarrolla con la herramienta *Simulink 3D Animation* de Matlab, el cual a través de *Simulink (Matlab)* permite visualizar la trayectoria de todo el sistema.

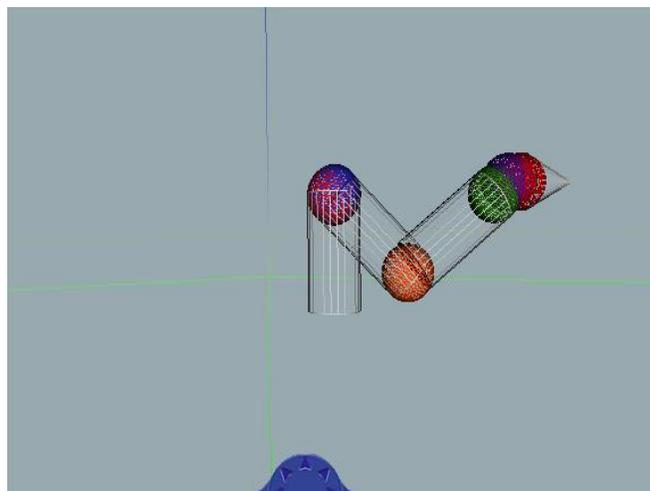


Figura 3. 3 Esquema Haptic 6DOF.

3.3 DISEÑO DE UN CONTROLADOR POR PAR CALCULADO (CTC)

El control por par calculado CTC (*Computed Torque Control*), utiliza el modelo dinámico inverso para linealizar y desacoplar la compleja dinámica del robot, asegurando teóricamente un comportamiento uniforme en cualquier configuración [18].

Para el diseño de este controlador es necesario saber la relación que existe entre la fuerza o cupla total, la matriz de inercia, los términos de Coriolis, fuerzas centrífugas y términos gravitacionales.

En la siguiente ecuación se muestra dicha relación.

$$\tau = A(q)w(t) + H(q, q') \quad \text{Ecuación 2.}$$

La explicación de las operaciones que se realizan posteriormente se encuentran explicadas en [18].

Adicionalmente es necesario sintonizar las ganancias proporcionales y derivativas del controlador lineal, K_p y K_v respectivamente. Las posiciones y velocidades deseadas se nombran como q^d y q'^d .

Con estos datos se desarrolla el modelo de control CTC, el cual posee un control lineal y un subsistema de desacoplamiento y compensación no lineal, además incluye el modelo dinámico (MD) del robot. Para efectos de simulación se agregó el MGI el cual recibe del joystick las posiciones deseadas y entrega al controlador las posiciones articulares.

En la Figura 3.4 se muestra el esquema de control CTC.

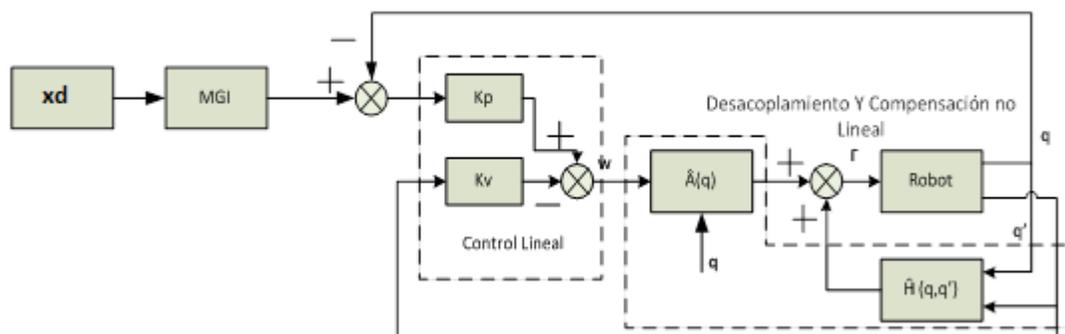


Figura 3. 4 Esquema del control CTC.

3.4 SINTONIZACIÓN DEL CONTROLADOR CTC.

La sintonización se realiza utilizando la metodología explicada en [18]. Luego de ingresar los valores de las distancias D3,R4 y los momentos de inercia en los modelos dinámicos, se procede a sintonizar las ganancias proporcionales y derivativas, con el fin de obtener un error pequeño.

En la Tabla 3. 6, se observa los respectivos valores de las ganancias que se generaron para obtener un error pequeño.

J	1	2	3	4	5	6
Kp	10000	9000	7000	7000	7000	7000
Kv	100	70	70	80	80	70

Tabla 3. 6 Ganancias del controlador

Con estos valores se determina el funcionamiento de la interfaz siguiendo una trayectoria circular en el plano x, y.

3.5 SEGUIMIENTO DE TRAYECTORIAS

Durante la simulación de la interfaz háptica, se utiliza una trayectoria circular, la cual posee como características un radio definido (5 cm), un tiempo de muestreo de 0.001 segundos y un punto de ubicación en los ejes x y y (0.01, 0.01).

En la Figura 3. 5, se observa el seguimiento de las trayectorias; mostrándose en verde la consigna circular de la interfaz y en azul la consigna circular deseada.

El error cartesiano de las trayectorias se observa en la Figura 3.6, donde el valor del error es aproximadamente $2,8 \times 10^{-4}$, el cual como resultado demuestra que el diseño del robot es capaz de seguir trayectorias simples.

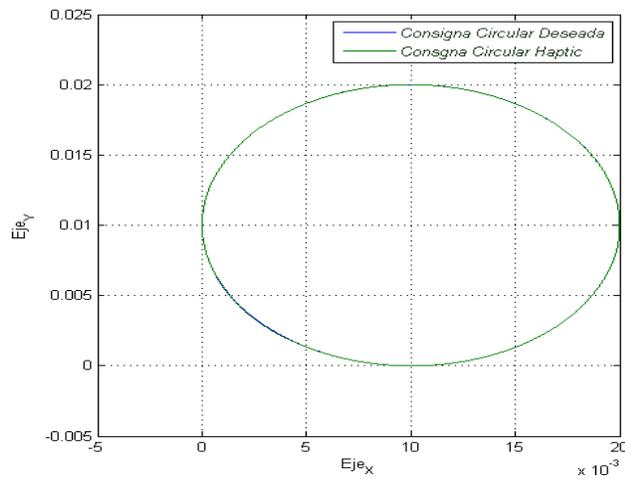


Figura 3. 5 Trayectoria circular.

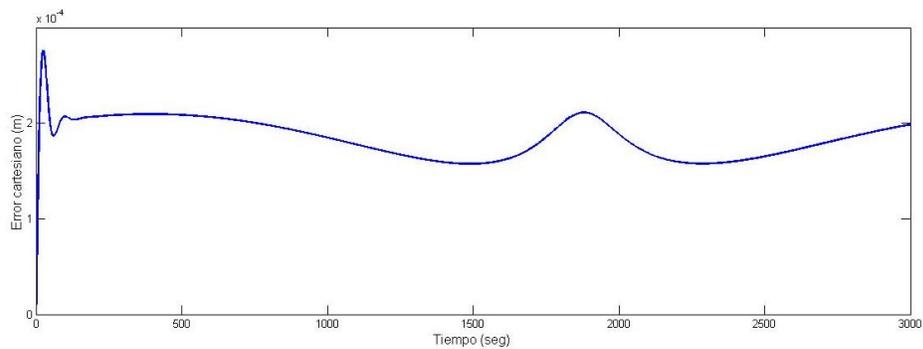


Figura 3. 6 Grafica del error cartesiano.

Finalmente en la Figura 3.7 se muestran los torques de cada articulación. Observándose que el máximo torque que se necesita es de 0.23 N*m o 2.35 kgf*cm. De esta forma se delimita el campo de escogencia del actuador principal; dando como mejor opción el servomotor digital Herkulex drs 001 el cual posee como máximo torque 12 kgf*cm. adicionalmente se puede comunicar en serie, disminuyendo el cableado de la interfaz. Por otro lado posee un software propio el cual permite configurar ciertas características del servo motor.

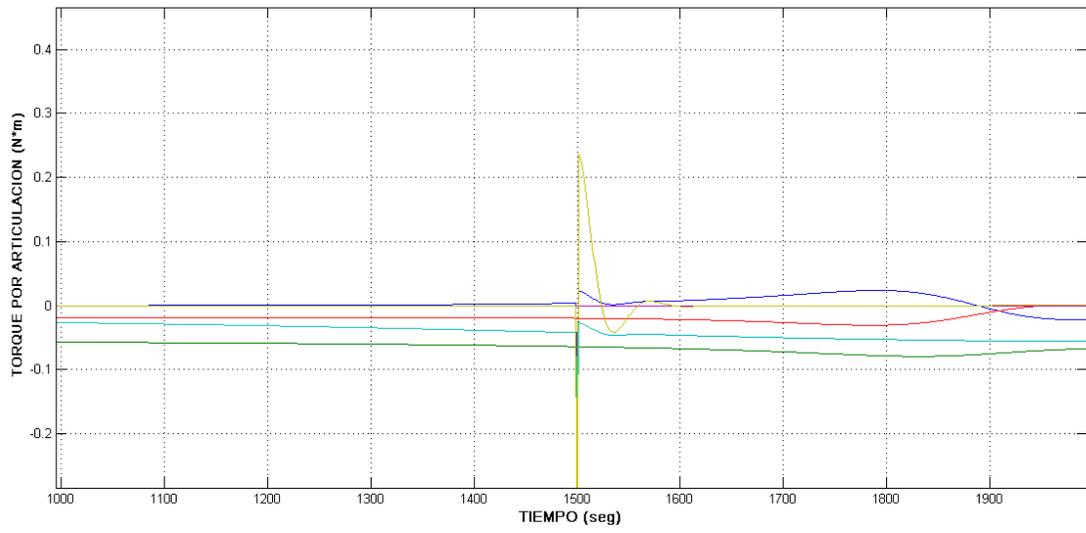


Figura 3. 7 Torque obtenido para cada articulación.

4 DISEÑO MECÁNICO Y CONSTRUCCIÓN DE HAPTIC 6DOF

En este capítulo se presenta el diseño de una interfaz con estructura en serie de seis grados de libertad.

El diseño de HAPTIC 6DOF permite obtener dimensiones, movimiento y geometría adecuados para su posterior construcción. Adicionalmente, mediante el diseño CAD se obtienen las propiedades físicas de cada material y se hallan los momentos de inercia, los cuales son necesarios para realizar la simulación.

4.1 DISEÑO CAD DE HAPTIC 6DOF

Este diseño se realizó gracias a la herramienta de diseño SolidEdge® con el propósito de obtener un acercamiento más preciso en el dimensionamiento de la interfaz. Las principales ventajas que se obtienen utilizando esta herramienta son:

- Crear un modelo análogo del prototipo, con el fin de caracterizar cada elemento en función de las propiedades; tales como forma, tamaño, material, etc.
- Permite analizar, ajustar y rediseñar las piezas directamente en el entorno del conjunto, con el fin de aproximarse a un prototipo adecuado, que tenga un correcto funcionamiento.
- Definir y simular los movimientos que genera cada una de las articulaciones, permitiendo determinar el espacio de trabajo. En la siguiente figura se muestra el diseño de HAPTIC 6DOF.

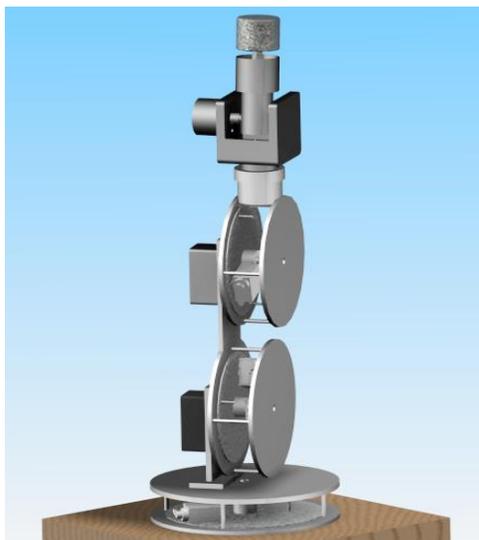


Figura 4. 1 Diseño de HAPTIC 6DOF.

En las siguientes figuras se muestra los diseños CAD de las piezas; los sensores y los actuadores tanto principales como secundarios.

En la Figura 4.2 Se muestra el actuador principal No 1 anclado a la base de la interfaz.



Figura 4. 2 Actuador principal No 1 anclado a base.

Para la construcción del cuerpo de rotación θ_1 se diseñaron 4 piezas, que acopladas permiten conectar al actuador principal No 1 con la base de la interfaz (Ver Figura 4. 3).

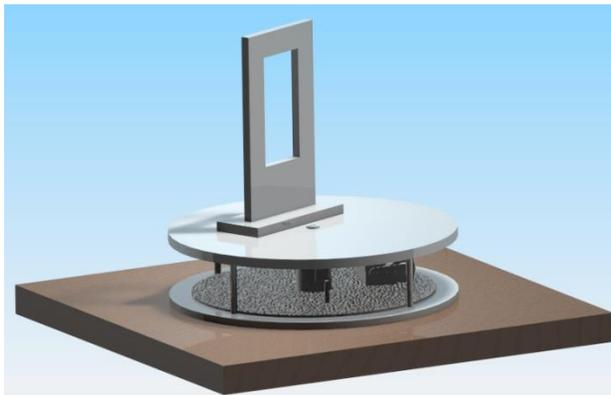


Figura 4. 3 Cuerpo de rotación θ_1 .

El cuerpo de rotación θ_2 está constituido por 4 piezas que se conectan con el actuador principal No 2, el cual se encuentra acoplado por medio de una pieza al cuerpo de rotación θ_1 (Ver Figura 4. 4).

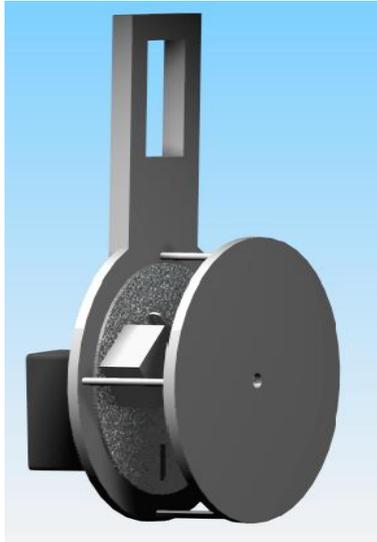


Figura 4. 4 Cuerpo de rotación θ_2 .

El cuerpo de rotación θ_3 está constituido por 4 piezas que se conectan con el actuador principal No 3, el cual se encuentra acoplado por medio de una pieza al cuerpo de rotación θ_2 . Adicionalmente este cuerpo de rotación también contiene el sensor de orientación No.1 en una de sus piezas (Ver Figura 4. 5).



Figura 4. 5 Cuerpo de rotación θ_3 .

El cuerpo de rotación θ_4 está formado por una sola pieza la cual se acopla al cuerpo de rotación θ_3 y adicionalmente contiene el sensor de orientación No. 2 (Ver Figura 4. 6)

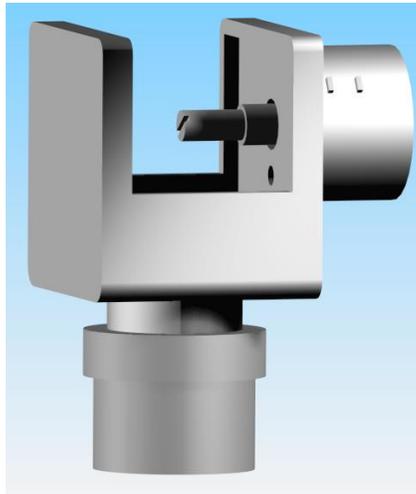


Figura 4. 6 Cuerpo de rotación θ_4 .

El cuerpo de rotación θ_5 está formado por una sola pieza y contiene el sensor de orientación No. 3 (Ver Figura 4. 7).

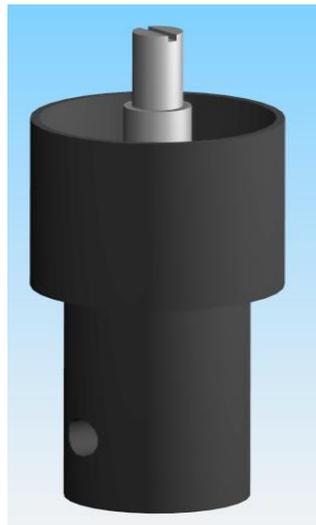


Figura 4. 7 Cuerpo de rotación θ_5 .

El cuerpo de rotación θ_6 está formado por una sola pieza la cual está ensamblada al sensor de orientación No. 3 (Ver Figura 4. 8).



Figura 4. 8 Cuerpo de rotación $\theta 6$.

4.2 SELECCIÓN DE MATERIALES Y COMPONENTES ELECTRÓNICOS PARA LA CONSTRUCCIÓN DE HAPTIC 6DOF.

Para la construcción de la interfaz se tuvieron en cuenta algunos datos obtenidos mediante la simulación y el diseño CAD. En primer lugar se determinó que las piezas deben formar una estructura en forma de exoesqueleto, con el fin de que sea más ergonómica, ligera y tenga una mejor presentación. Adicionalmente este tipo de estructura permite disminuir el peso que deben soportar los servomotores.

Con la simulación logramos determinar las cuplas, permitiendo definir los posibles actuadores que se iban a utilizar en la construcción de la interfaz. A continuación se muestra con más detalle el material y los componentes que se utilizaron en la construcción de HAPTIC 6DOF.

4.2.1 Material para la construcción de las piezas.

Gracias al acercamiento del modelo físico de HAPTIC 6DOF, obtenida por medio del software CAD SolidEdge® y observando el material de otro tipo de interfaces que se encuentran en el mercado; se determinó que los materiales adecuados para la construcción de la interfaz son:

- Polipropileno (acrílico), el cual presenta las siguientes características:
 - Resistencia a la intemperie.
 - Excelentes cualidades para el mecanizado y el termo doblado.
 - Peso ligero y de baja densidad.
 - Alta resistencia al impacto.
 - Dureza similar al aluminio.

- Aluminio, que presenta las siguientes características:
 - Ligero y de baja densidad
 - De fácil mecanizado.
 - Muy maleable y dúctil.
 - Material soldable.
 - Resistencia a la abrasión y al desgaste.
 - Resistencia a la corrosión.
 - No necesita mantenimiento
 - Acabado decorativo ya que se pueden obtener diferentes tonalidades.

Se utilizan estos dos materiales con el fin de construir dos tipos de placas de distinto grosor. Para la primera placa se necesita una dureza considerable con un grosor aproximado de 3 mm; escogiéndose el acrílico como material de construcción de estas piezas. Para la segunda placa se necesita mayor dureza, ya que se debe soportar las fuerzas ejercidas por los resortes. Además de tener baja densidad con el fin de generar poca fricción entre las dos placas. Por tanto el material que se escogió fue el aluminio con un grosor de 1 mm.

- Mdf, que presenta las siguientes características:
 - Resistente a la alta temperatura y la humedad.
 - Tamaño de fibra homogéneo en todo el espesor.
 - Sólido y compacto
 - Sus fibras permiten precisión en el corte y, en el momento de atornillar.
 - Superficie plana y tersa.

Este último material fue utilizado solamente para la base de la interfaz, ya que era necesario un peso adecuado, que soportara toda el peso de HAPTIC 6DOF.

4.2.2 Selección de actuadores.

Para esta selección se tuvieron en cuenta los siguientes aspectos:

- Cuplas obtenidas a partir de la simulación, las cuales oscilan en los 4 Newton/metro.
- Suplir la necesidad de proporcionar un par adecuado para el frenado de un eslabón (actuador principal).

- Permitir el anclaje preciso entre las dos placas que poseen los tres primeros cuerpos de rotación (actuador secundario).
- Los diferentes tipos de actuadores utilizados en aplicaciones robóticas y que se encuentren disponibles en el mercado nacional.
- **Actuador principal.** Teniendo en cuenta estos aspectos se determinó que el servo motor que se adecua mejor a estas exigencias es el servo digital HERKULEX DRS 0101 que se muestra en la siguiente figura.



Figura 4. 9 Servo digital: HERKULEX DRS 0101.

Este servo tiene las siguientes características:

- Capacidad de controlar su velocidad y posición, temperatura, carga, voltaje, etc.
- El algoritmo de control de posición de cada servo puede ajustarse individualmente.
- Angulo de giro controlable entre 0 y 300°, con posibilidad de funcionamiento como motor de rotación continua.
- Comunicación serial full- dúplex que permite conectar el servomotor directamente a sistemas de control como arduino.
- Velocidad: 0.166sec/60° a 7.4V.
- Torque 12kg.cm a 7.4V.
- Peso: 45 g.

Este servo tiene la capacidad de conectarse en serie con los demás servos (Ver Figura 4.10) (a) y su distribución de pines (b)

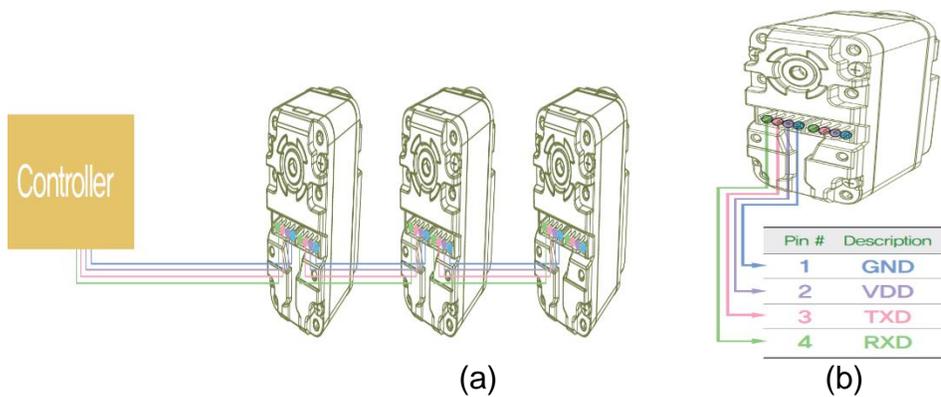


Figura 4. 10 Conexión en serie (a) y distribución de pines (b)

- **Actuador secundario.** Este es un servo análogo H-KING 15178 que por su dimensión y especificaciones se ajusta a los requerimientos y al diseño de la interfaz (Ver Figura 4. 11)



Figura 4. 11 Servo análogo H-KING 15178.

Este servo posee las siguientes características:

- Torque: 1.2kg/cm @ 4.8v, 1.4kg/cm @ 6v.
- Peso: 10g.
- Velocidad: 0.10/60deg @ 4.8v, 0.09/60deg @ 6v.
- Voltaje: 4.8v~6v.
- Dimensiones: (L) 32 mm * (W) 12mm * (H) 30 mm.

4.2.3 Selección de sensor.

Para este proyecto se utilizó un sensor, que de acuerdo al tipo de articulación en el que se ubica, recibe el nombre de sensor secundario para las articulaciones de orientación y sensor de posición para las articulaciones de posición.

Este sensor es un potenciómetro logarítmico; utilizado en las primeras 3 articulaciones con el fin de determinar la posición de una de las placas y así generar el bloqueo cuando se requiera. Adicionalmente se le añade un condensador de tantalio con el propósito de filtrar señal dc. Mientras que en las siguientes articulaciones permite determinar la orientación que desea el usuario.



Figura 4. 12 Potenciómetro logarítmico.

Presenta las siguientes características:

- Resistencia: 1k ohm.
- Potencia: 1/2 W - 0.5 W - 500 mW.
- Tipo: Lineal rotativo 16 mm cuerpo.
- Diámetro de eje: 6 mm - 1/4".

4.2.4 Hardware adicional.

- **Resorte.** El muelle utilizado pertenece a una variedad de acero llamada acero elástico, el cual presenta alta flexibilidad, adicionalmente tiene la capacidad de absorber la energía cinética de una forma gradual y eficiente.

Para entender la función que cumplen los resortes dentro de las articulaciones primero se muestran las dos placas ancladas gracias al actuador secundario; lo cual indica que se mueven conjuntamente y por tanto no existe elongación (Ver figura 4.13).

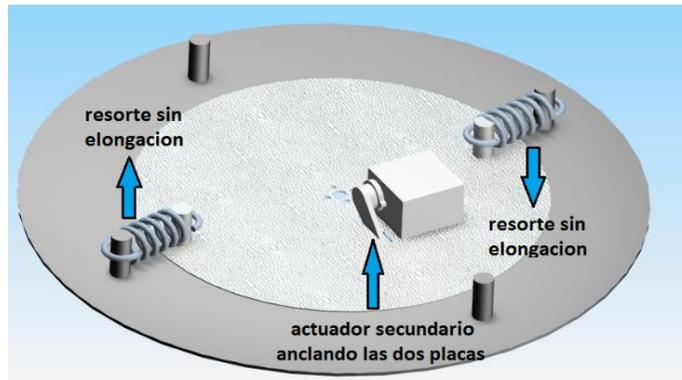


Figura 4. 13 Placas ancladas.

Posteriormente una vez se presenta el evento de colisión, el actuador secundario desancla las dos placas generándose una resistencia por medio del resorte ante la fuerza ejercida por el usuario.

En la Figura 4.14 se muestra la elongación que se presenta en el resorte, cuando no están ancladas las dos placas.

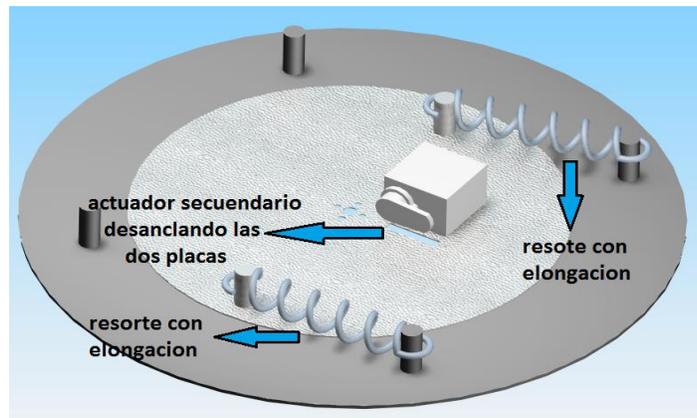


Figura 4. 14 Placas desancladas.

A continuación se presentan las características del resorte que se utilizó (Ver Figura 4. 15):

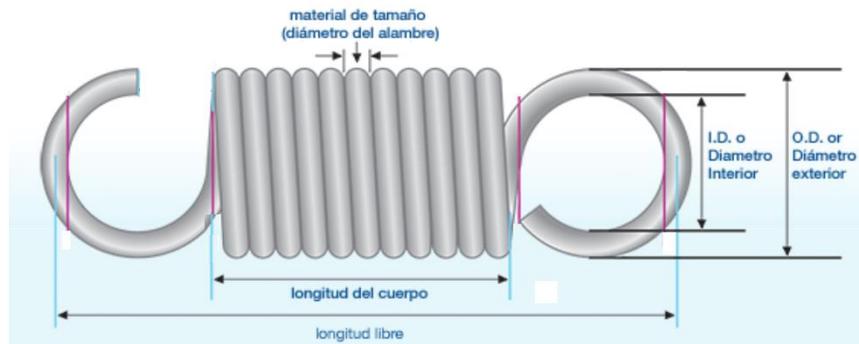


Figura 4. 15 Características generales del Resorte.

- Diámetro interior: 6 mm
- Diámetro exterior: 7,5 mm
- Diámetro del alambre: 1,2 mm
- Longitud del cuerpo: 13 mm
- Longitud libre: 23 mm

En la tabla siguiente se muestran los valores de las masas y la elongación del resorte, con el fin de determinar la constante del resorte; según la ley de Hooke [28], donde $F = -Kx$.

No	Masa (gramos)	ΔX (mm)
1	250	4
2	300	5
3	350	6
4	450	8
5	550	11
6	650	13
7	750	16
8	850	19
9	950	21

Tabla 4. 1 Datos obtenidos en laboratorio.

Teniendo los anteriores datos, se ingresan a Matlab, el cual genera la siguiente gráfica:

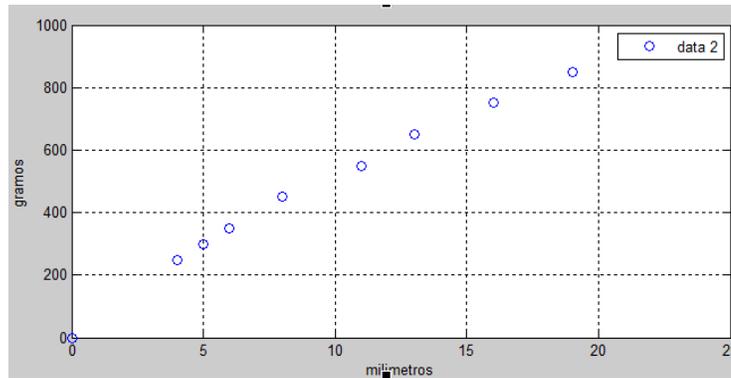


Figura 4. 16 Mapeo de datos obtenidos en laboratorio.

Dado que los datos no son lineales, se utiliza una función de Matlab para linealizar y poder hallar la pendiente exacta de la recta generada. En la Figura 4. 17 se muestra la gráfica obtenida.

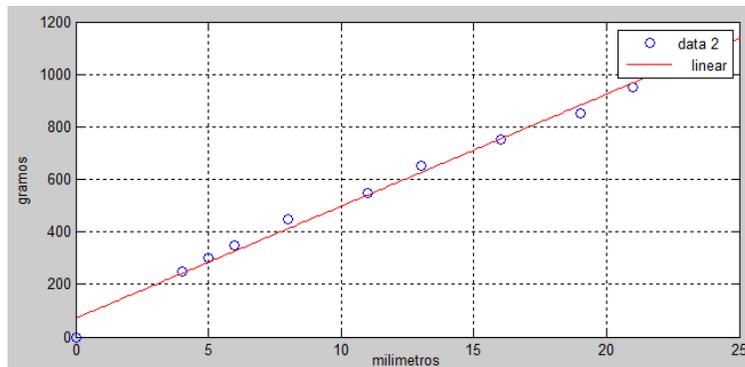


Figura 4. 17 Linealización de los datos.

Los resultados que se obtienen de la recta son:

$$Y = P1 * X + P2$$

Con coeficientes: P1 = 42.677 y P2 = 70.427

Por tanto la pendiente obtenida es: 42.677 gr/ mm

Para tener una mejor idea sobre la constante del resorte, hacemos una conversión de unidades.

$$42.677 \left(\frac{gr}{mm} \right) \left(\frac{1kg}{1000gr} \right) \left(\frac{10 mm}{1 cm} \right) = 0.42677 \left(\frac{kgf}{cm} \right)$$

Debido a que en cada una de las articulaciones de posicionamiento se utilizan dos resortes en paralelo, las constantes se deben sumar.

$$K_t = k_1 + k_2, \text{ por tanto } K_t = 0,85354 \left(\frac{\text{kgf}}{\text{cm}} \right)$$

En el capítulo 4 en la sección de construcción de realimentación se explica la relación que se va a realizar entre la fuerza de animación, la constante del resorte y su elongación.

- **Regulador de voltaje.** Es una fuente/ convertidor DC *step down* o reductor de voltaje, de alto rendimiento que utiliza un chip LM2596 que se adecua a los requerimientos de alimentación para los servos digitales HERKULEX DRS 0101 y los micro servos H-KING 15178. Se muestra en la Figura 4. 18

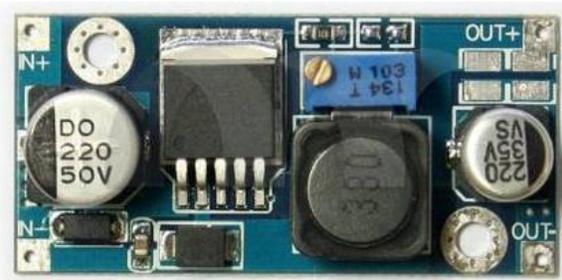


Figura 4. 18 Regulador de voltaje.

Presenta las siguientes características:

- Voltaje de entrada 4.5 a 35 V
- Voltaje de salida 1.25 a 30 V(ajustable)
- Corriente de salida: 2A nominal, 3A máximo (añadir el disipador de calor)
- Protección contra cortocircuitos: limitación de corriente.
- Tamaño 48 (L) * 23 (W)* 14 (H) mm (con potenciómetro).
- **Fuente de alimentación:** la fuente seleccionada cumple con los requerimientos de la interfaz ya que permite



Figura 4. 19 Fuente de alimentación.

- **Tarjeta de expansión:** este dispositivo electrónico nos permite tener una conexión ordenada con los diferentes componentes que intervienen en el funcionamiento de la interfaz. Una vez se determinaron los elementos se procedió a su diseño mediante el software Eagle, el cual nos permite una mejor integración de los dispositivos electrónicos. Tanto el diseño PCB como el diseño esquemático de la tarjeta se muestran en el ANEXO B.

En la Figura 4.20 se muestra la tarjeta de expansión.



Figura 4. 20 Tarjeta de expansión.

4.2.5 Hardware de comunicación.

Para este proyecto se buscó una plataforma hardware que permitiera simplificar el proceso de trabajar con microcontroladores, que fuera asequible, tuviera un entorno de programación simple y directo, con código abierto que ayudara a cumplir con las necesidades de transmisión y recepción de los datos que se generan desde y hacia la interfaz.

- Tarjeta de adquisición Arduino

Esta plataforma gracias a su robustez, fácil programación y sus múltiples entradas - salidas digitales y analógicas, permiten leer y controlar gran variedad de dispositivos físicos, lo cual es indispensable para este proyecto debido a que se tienen múltiples sensores y actuadores.

Otra ventaja que presenta los proyectos realizados con Arduino es su posibilidad de comunicación con otro tipo de software, en este caso con el software encargado del entorno 3D.

Presenta las siguientes características

- Microcontroladores : ATmega2560.
- Tensión de funcionamiento: 5V.
- Voltaje de entrada (recomendado): 7-12V.
- Voltaje de entrada (límites): 6-20V.
- Digital I / O Pins: 54 (de los cuales 15 proporcionan salida PWM).
- Pines de entrada analógica: 16.
- Corriente continua para las E / S Pin: 40 mA.

Se muestra en la Figura 4. 21.



Figura 4. 21 Tarjeta ATmega2560.

4.3 ARTICULACIONES HAPTIC 6DOF

Este primer prototipo de interfaz activa, tiene características antropomórficas debido a que las 3 primeras articulaciones simulan un sistema (hombro - codo) las cuales permiten posicionar el objeto a manipular en el entorno 3D. Posteriormente las 3 últimas articulaciones permiten orientar el objeto en cualquiera de los 3 ejes.

4.3.1 Articulaciones de posicionamiento.

Estas primeras articulaciones están construidas con los siguientes elementos: actuador principal, resortes, actuador secundario, sensor secundario, placas de acrílico, placa de aluminio. El funcionamiento de estas articulaciones se explicará más adelante.

En la primera articulación el actuador principal está asegurado a la base como se muestra en la Figura 4.22 (a). Para soportar el peso de la interfaz se hizo necesario construir una base de MDF con dimensiones 19.5 cm (largo)* 19.5 cm (ancho)* 9 cm (alto). En la Figura 4.22 (b) se muestra la vista frontal de la articulación.

En esta articulación la placa de aluminio tiene un radio de 11cm, con un grosor de 1 mm. La placa de acrílico tiene 13 cm de radio con un grosor de 3 mm.

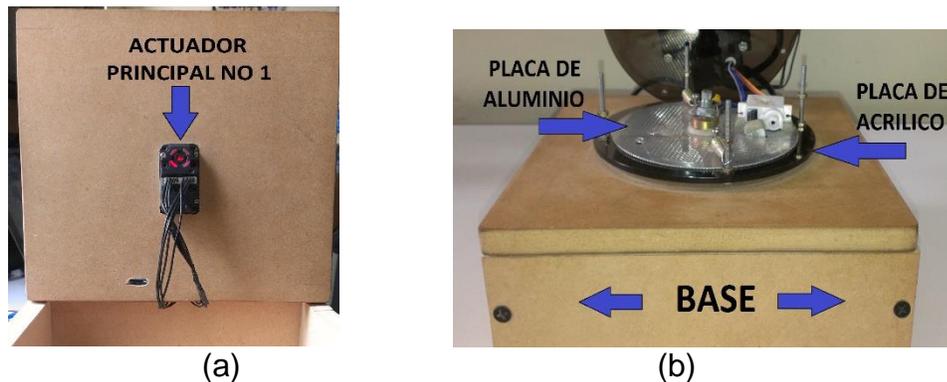


Figura 4. 22 Vista inferior del actuador (a) y vista frontal de la articulación (b).

La segunda articulación está anclada a la primera mediante un soporte de acrílico, el cual tiene sujeto al actuador principal No 2., como se muestra en la Figura 4. 23

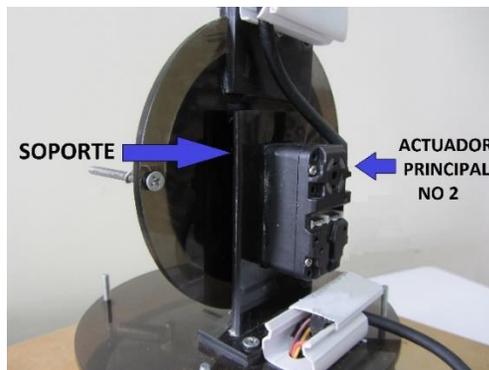


Figura 4. 23 Soporte de la segunda articulación.

En esta segunda articulación la placa de aluminio mide 9 cm de radio con grosor de 1 mm. La placa de acrílico tiene 11 cm de radio con un grosor de 3 mm (Ver Figura 4.24 a.), adicionalmente esta placa de acrílico tiene un soporte rectangular al cual está sujeto el tercer actuador principal (Ver Figura 4.24 b.).

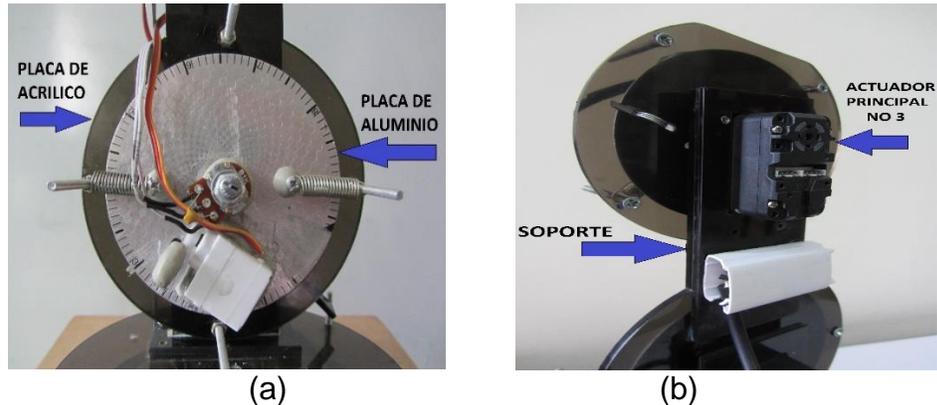


Figura 4. 24 Placas de segunda articulación (a) y soporte de acrílico para tercer actuador (b).

En la tercera articulación la placa de aluminio mide 8,4 cm de radio con un grosor de 1mm. La placa de acrílico mide 10,5 cm de radio con un grosor de 3mm (Ver Figura 4.25 a.), adicionalmente en esta articulación se encuentran ancladas las siguientes 3 articulaciones de orientación gracias a una base de acrílico con rieles; las dimensiones de la base son: 4,8 cm (largo) * 4 cm (ancho) * 6 mm (alto) (Ver Figura 4.25 b.).

Estos rieles se acoplan a las placas de acrílico formando un soporte para las ultimas 3 articulaciones (Ver Figura 4.25 a.).

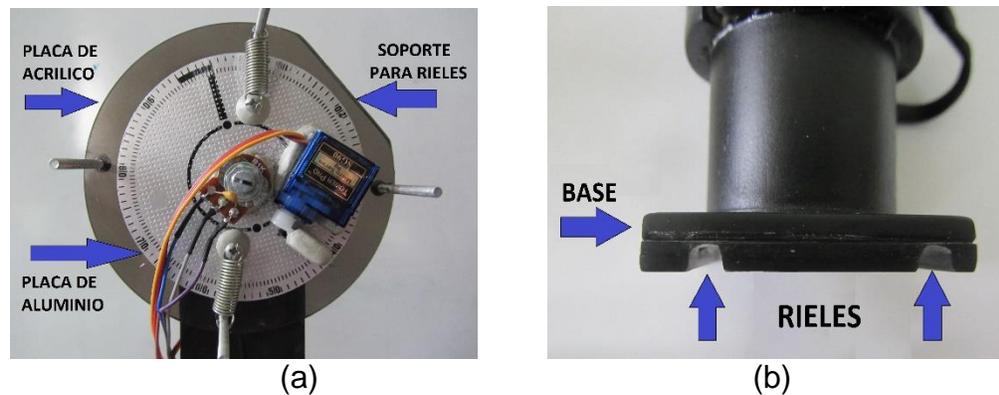


Figura 4. 25 Placas (a) y rieles del tercer actuador (b).

En la Figura 4. 26 se muestra los componentes que tiene cada articulación de posicionamiento. Posteriormente en la Figura 4.27 se muestra la dirección de las dos placas ancladas por el actuador secundario; las cuales se mueven conjuntamente debido a la fuerza que ejerce el usuario.

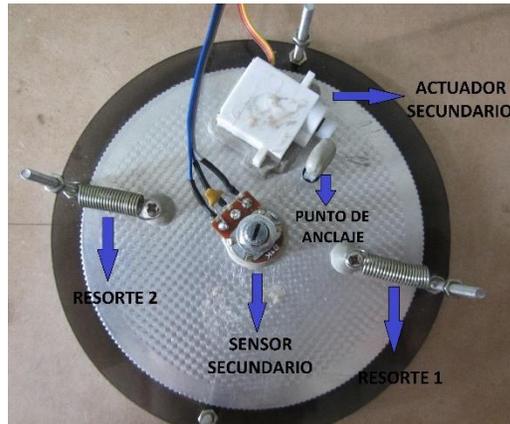


Figura 4. 26 Componentes de las articulaciones de posicionamiento.



Figura 4. 27 Dirección de las fuerzas que actúan en ausencia de colisiones.

Posteriormente cuando se presenta una colisión, el actuador secundario desancla las placas y se genera una fuerza de realimentación; que es contraria a la fuerza ejercida por el usuario (Ver Figura 4. 28).



Figura 4. 28 Dirección de las fuerzas que actúan con colisión.

A continuación se muestra la imagen de las 3 articulaciones de posicionamiento en conjunto

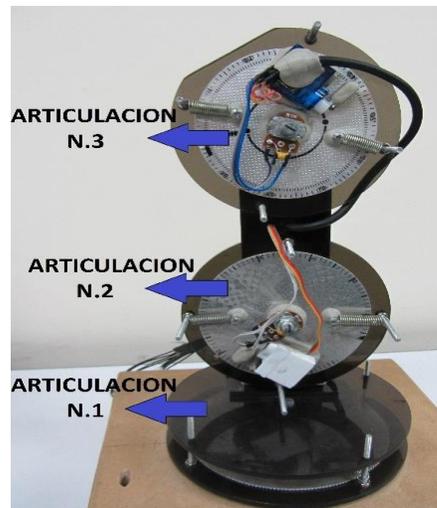


Figura 4. 29 Articulaciones de posicionamiento.

4.3.2 Articulaciones de orientación.

Estas últimas 3 articulaciones están construidas con piezas cilíndricas de PVC que contienen los sensores de orientación y piezas de acrílico que forman su estructura.

La cuarta articulación tiene una base que soporta un contenedor cilíndrico dentro del cual está el sensor de orientación No. 1. El eje del sensor está anclado a un

disco de acrílico que se encuentra dentro de unos cilindros de rotación (Ver Figura 4. 30).

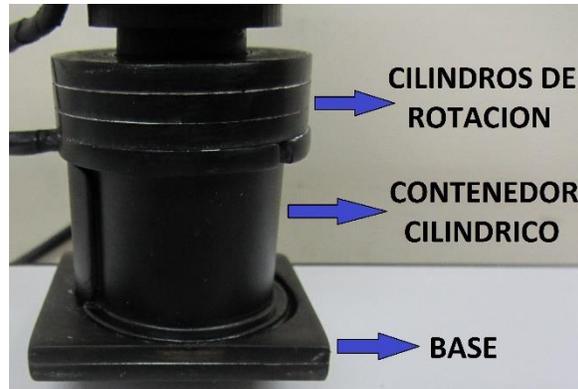


Figura 4. 30 Piezas cuarta articulación.

La quinta articulación tiene un contenedor cilíndrico donde se encuentra el sensor de orientación No 2. Este contenedor se encuentra asegurado a una pieza de acrílico que a su vez sirve como soporte para la última articulación (Ver Figura 4 31).



Figura 4. 31 Piezas quinta articulación.

La sexta articulación está compuesta por 3 piezas. La primera es una pieza de PVC que está asegurada al eje del sensor de orientación No. 2 en la parte inferior. Mientras que la parte superior sostiene al contenedor cilíndrico dentro del cual se encuentra el sensor de orientación No 3. Finalmente al eje de este sensor está unida una pieza de rotación (Ver Figura 4. 32).

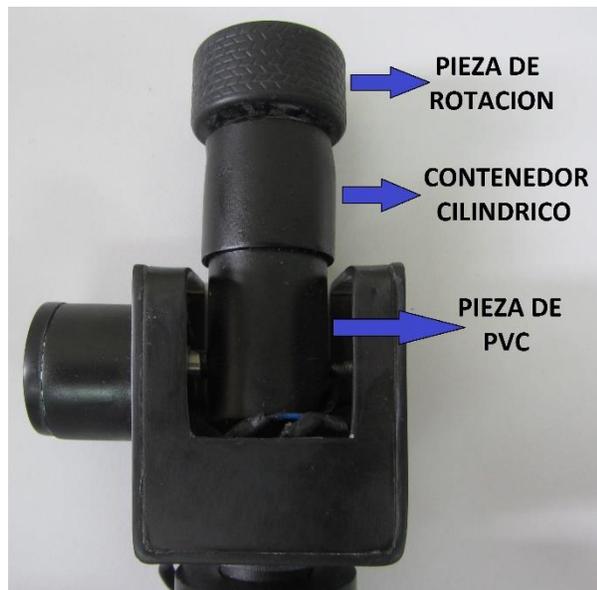


Figura 4. 32 Piezas sexta articulación.

A continuación se muestran las articulaciones de orientación en conjunto. (Ver Figura 4. 33)

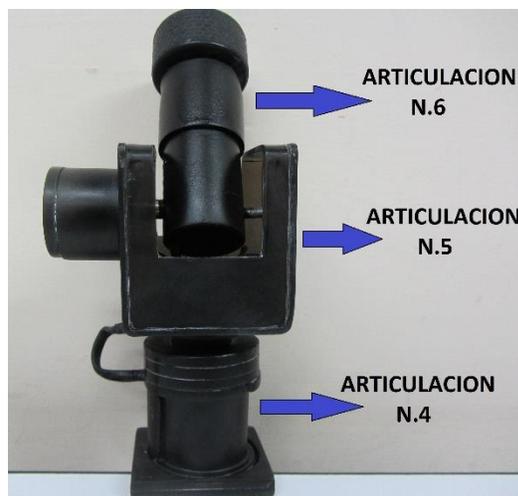


Figura 4. 33 Articulaciones de orientación.

4.4 CONSTRUCCIÓN DEL SISTEMA DE REALIMENTACIÒN

Las tres primeras articulaciones funcionan gracias a un actuador principal anclado a una placa de aluminio; también posee dos piezas de acrílico que reciben la fuerza del usuario para ser trasferida por medio de dos resortes a la placa de

aluminio. El sensor secundario ayuda a determinar la magnitud y dirección del giro una vez se presente el evento de colisión.

Cuando la colisión se presenta, la unión entre el actuador y la placa de aluminio permitirá ejercer una fuerza contraria a la fuerza ejercida por el usuario, que en este caso se llamará fuerza de restitución.

El actuador secundario tiene dos funciones, la primera es anclar las placas de aluminio y acrílico mientras no se esté tocando ninguna zona de colisión, la segunda es desanclar las placas una vez el usuario haya tocado alguna de las zonas de colisión.

El entorno donde se moverá el objeto móvil en este proyecto será un cubo (objeto estático), por tanto el evento de colisión se presentara cada vez que el usuario toque cualquier cara del cubo, inicialmente el objeto a manipular por el usuario será una esfera.

Para entender el funcionamiento general del sistema, se dividirá este en dos eventos.

El primero cuando el movimiento del efector esté dentro de las paredes del cubo y el segundo cuando se presente una colisión y se necesite generar una realimentación de fuerza.

- **Primer evento.** En el primer momento, debido a que no se requiere realimentación de fuerzas, las placas de aluminio y acrílico se moverán conjuntamente, gracias a que el actuador secundario las mantiene ancladas. En el siguiente esquema se muestra el área de desplazamiento sin colisiones (área sombreada azul), utilizando x como eje de movimiento y $P(x)$ como variable. Por simplicidad no se analizan los movimientos en los ejes y e z .

$P(x)$ está entre los valores x_0 y x_1 (Ver Figura 4. 34).

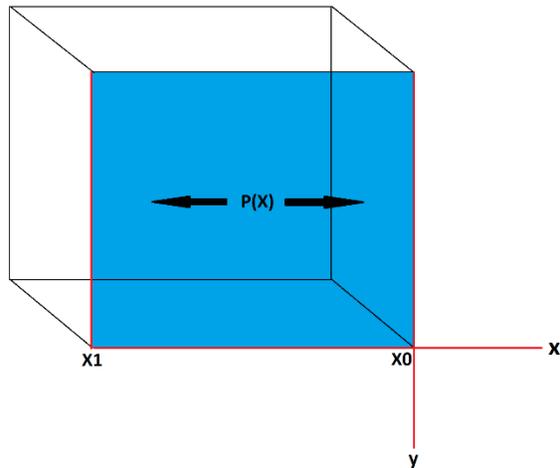


Figura 4. 34 Diagrama primer evento.

- **Segundo evento.** En el segundo momento, debido a que el usuario ha tocado una zona de colisión, el actuador secundario desancla las placas, permitiendo que el actuador principal que se encuentra unido con la placa de aluminio ejerza una fuerza de restitución contraria a la ejercida por el usuario. En este momento el sensor secundario entrega la posición de $P(x)$. En el esquema que se ve a continuación (Ver Figura 4. 35), se muestra el área sobre la cual se ejerce una realimentación de fuerzas (área sombreada verde), donde $P(X)$ está entre los valores de X_0 y X_F .

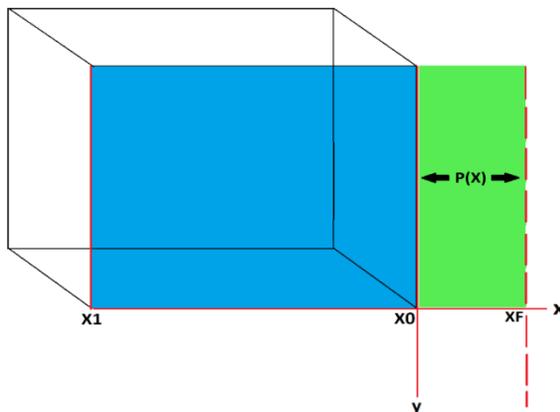


Figura 4. 35 Diagrama segundo evento.

La zona sombreada en verde está dividida en tres tipos de realimentación como se muestra en el siguiente diagrama (Ver Figura 4. 36).

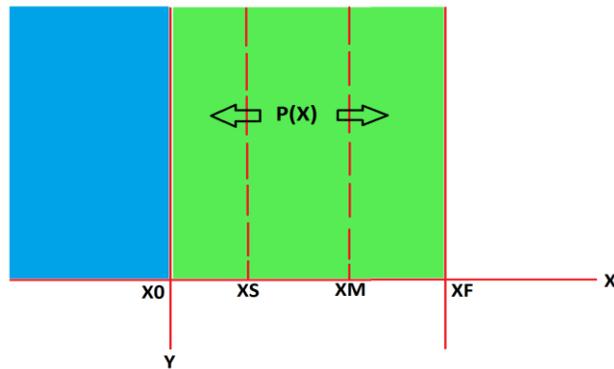


Figura 4. 36 Tipos de realimentación.

Para este proyecto se definieron 3 tipos de realimentación los cuales se diferencian de la siguiente forma:

- Realimentación suave (XS) = se simulará mediante un ángulo de giro 10 grados realizado por el actuador principal y contrario a la fuerza ejercida por el usuario.
- Realimentación Media (XM) = se simulara mediante un ángulo de giro 20 grados realizado por el actuador principal y contrario a la fuerza de usuario.
- Realimentación fuerte (XF) = se simulara mediante un ángulo de giro 30 grados realizado por actuador principal y contraria a la fuerza de usuario.

Teniendo estos ángulos de giro predefinidos, se realiza una relación con la elongación producida por el resorte, como se muestra en la tabla

Teniendo los datos Kt y ΔX , utilizamos la ley de Hooke [28], donde la fuerza es igual:

$$F = Kt * \Delta X$$

Esto con el fin de conseguir una aproximación de la fuerza que debe realizar el usuario cuando deforma un objeto en el ambiente real. Para realizar dicha aproximación se asemejará que la fuerza de la fórmula es igual a una fuerza de animación. Cabe aclarar que la fuerza obtenida en la fórmula es distinta a la $F_{animación}$ ya que esta última vincula dos constantes de elongación. Adicionalmente la $F_{animación}$ se compone por dos tipos de fuerza, la primera generada por el usuario en el momento que exista colisión y la segunda realizada por el actuador principal y en dirección contraria a la del usuario.

Por tanto tenemos que: $F_{anim} = (K1 + K2) * \Delta X$

Como se explicó anteriormente para este proyecto se predefinieron 3 tipos de realimentación, las cuales se obtuvieron realizando primero una comparación entre los ángulos generados por el servo y los valores de la elongación del resorte.

Angulo servo (θ°)	$\Delta x(\text{mm})$
0	0
10	3
20	4
30	3
40	3
50	3
60	4

Tabla 4. 2 Relación entre ángulo v/s elongación del resorte.

El segundo paso fue realizar una comparación entre los ángulos generados por la fuerza del usuario y los valores enteros entregados por el potenciómetro, a continuación se muestra la tabla de esta comparación que se realizó con la pared derecha del cubo.

Angulo (θ°) generado por usuario.	Lecturas potenciómetro
0	125
10	133
20	143
30	153
40	161
50	168
60	177

Tabla 4. 3 Relación entre ángulo v/s lecturas potenciómetro.

Con estos datos se puede diferenciar entre los valores que debe entregar el potenciómetro gracias a la fuerza ejercida por el usuario y el ángulo que debe hacer el actuador principal en dirección contraria a la del usuario.

Ya que el propósito de estas relaciones es lograr simular el comportamiento de deformar un objeto en el ambiente real; primero se hace una lectura de la fuerza que realiza el usuario por medio del sensor secundario, con este valor se obtiene el ángulo de giro correspondiente a dicha fuerza. Posteriormente para guardar una

equivalencia, el actuador principal realiza un ángulo de giro igual al realizado por el usuario. A continuación se explican los tipos de realimentación con más detalle.

Para la realimentación suave se debe tener en cuenta que el valor inicial entregado por el potenciómetro es de 125, luego de presentarse un evento de colisión. Por tanto para la realimentación suave el usuario debe llegar al valor de 133 en la lectura del potenciómetro, que equivale a 10 grados (ver tabla 4.2). Seguidamente el actuador principal hace un giro de 10 grados contrario al del usuario. En conclusión entre la fuerza ejercida por el usuario y el ángulo dado por el actuador se obtiene un ángulo total de 20 grados.

Para la realimentación media el usuario debe llegar al valor de 153 en la lectura del potenciómetro o 10 grados adicionales a la primera realimentación (ver tabla 4.2). En este punto el actuador principal hace un giro contrario de 10 grados, logrando en total un ángulo de 40 grados entre la fuerza del usuario y la realizada por el actuador

Por ultimo para la realimentación fuerte el usuario debe llegar al valor de 168 en la lectura del potenciómetro o 10 grados adicionales a la realimentación media. En este punto el actuador principal hace un giro contrario de 10 grados. Obteniendo en total 30 grados hechos por el usuario y 30 grados generados por el actuador.

En la siguiente tabla se muestran la relación que existe entre: los ángulos del servo, las lecturas del potenciómetro y la elongación del resorte.

Angulo servo (θ°)	Lecturas potenciómetro	(xi)Longitud resorte(mm)	Δx (mm)
0	125	20	0
10	133	23	3
20	143	27	4
30	153	30	3
40	161	33	3
50	168	36	3
60	177	40	4

Tabla 4. 4 Relación entre datos actuador principal-sensor secundario-resorte.

Con los datos que se muestran en la tabla 4.3, podemos hallar cuales son las fuerzas de animación que se deben realizar en los tres tipos de realimentación. Adicionalmente para cada fuerza de animación se predeterminan 3 colores, con el fin de ser diferenciadas en el entorno 3D; en el capítulo de resultados se muestran con más detalle.

- F_animación 1 = realimentación suave = Amarillo
- F_animación 2 = realimentación media = Naranja
- F_animación 3 = realimentación fuerte = Rojo

Para obtener las fuerzas de animación, primero se debe determinar cuál es la elongación del resorte; Dicha elongación se obtiene dependiendo del tipo de realimentación que se esté efectuando, posteriormente se multiplica la suma de las dos constantes de los resortes. A continuación se muestran las operaciones realizadas para obtener las fuerzas de animación antes mencionadas.

Utilizando la fórmula: $F_{\text{animación}} = (K1 + K2) * \Delta X$

- $F_{\text{animación 1}} = (0,85354 \left(\frac{\text{kgf}}{\text{cm}}\right)) * (0,7\text{cm})$

$F_{\text{animación 1}} = 0,59745 \text{ kgf.}$

- $F_{\text{animación 2}} = (0,85354 \left(\frac{\text{kgf}}{\text{cm}}\right)) * (1,3\text{cm})$

$F_{\text{animación 2}} = 1,109602 \text{ kgf}$

- $F_{\text{animación 3}} = (0,85354 \left(\frac{\text{kgf}}{\text{cm}}\right)) * (2,0\text{cm})$

$F_{\text{animación 3}} = 1,707 \text{ Kgf}$

Cabe aclarar que estos procedimientos se realizan buscando una aproximación de realimentación. No necesariamente son los procedimientos aptos para asemejar el comportamiento real de un material.

En la sección de trabajos futuros, se aclara como se puede realizar otros tipos de realimentación, con el fin de mejorar el funcionamiento de la interfaz.

5 HERRAMIENTAS SOFTWARE

En el desarrollo de la interfaz háptica (HAPTIC 6DOF), se utilizaron varias herramientas software para el diseño, simulación, calibración y control del dispositivo.

5.1 SOFTWARE DE CONTROL

5.1.1 Herkulex manager

Para el control mecánico de la HAPTIC 6DOF, se inicia con el proceso de identificación de los motores, los cuales cumplen ciertas características para el funcionamiento del sistema. Este tipo de motores entregan un conjunto de datos válidos para la tarjeta de control, en este caso la tarjeta ATmega 2560.

Para la configuración de este tipo de servomotores existe una aplicación mediante la cual se puede realizar los ajustes de cada uno de los servomotores, además permite observar la lectura de la posición en voltaje (PWM), o la posición en ángulos entre otras características (Ver Figura 5.1). En este caso para la realización de las primeras pruebas se utilizó la configuración que estaba por defecto.

El Kit HerkuleX Manager proporciona una manera eficiente de conexión y comunicación para el servomotor inteligente HerkuleX [19], ya que primero permite que los comandos sean reconocidos directamente con la plataforma ARDUINO. El segundo aspecto relevante está en la señal de control, la cual es enviada mediante un paquete de solicitud y recepción de datos ACK (*acknowledgement* o acuse de recibo o asentimiento), y por último dado que los servomotores están conectados en serie, se necesita una correcta identificación de los paquetes que son enviados a cada uno de ellos.

Su instalación se puede ver en los anexos con sus respectivos drivers.



Figura 5. 1 Software Herkulex Manager.

5.1.2 Plataforma Arduino

La plataforma ARDUINO toma la información del entorno a través de sus pines de entrada entre una múltiple gama de sensores, además puede afectar aquello que le rodea controlando luces, motores y otros actuadores [20]. Es por eso que este dispositivo es ideal para el control de la interfaz HAPTIC 6 DOF.

Para el control de la HAPTIC 6DOF se utiliza la tarjeta ARDUINO MEGA 2560. De esta tarjeta se utilizan: sus entradas y salidas analógicas-digitales para las conexiones de los sensores y actuadores, la memoria EEPROM para almacenar el dato del punto de colisión, módulo de transmisión y recepción de datos para interactuar con la interfaz y comunicación serial para comunicarse con el entorno 3D. Con las anteriores características se concluye que esta tarjeta es apta para el control del sistema, además de que posee una cómoda forma de programación.

Adicionalmente esta plataforma permite la vinculación de los servos digitales directamente, adicionando sus propias librerías. Con este recurso se puede conocer y aplicar las diferentes funciones de los servos, además de reconocer e interpretar su estado de funcionamiento (posición angular, temperatura, estado del servomotor).

5.2 SOFTWARE DE INTERFAZ DE USUARIO

La interfaz de usuario es el medio con que el usuario puede comunicarse con el dispositivo háptico, a través de la computadora, ya sea interactuando por medio de botones, barras de desplazamiento o recibiendo información por medio de ventanas [21].

En el análisis del funcionamiento de la interfaz háptica se escoge un software que permita desarrollar múltiples funciones, además de que pueda interactuar con diferentes tipos de software y entrelazarlos.

Para el desarrollo de esta aplicación se deben tener en cuenta algunas características del software a utilizar, algunas de estas son:

- Obtener un software accesible, que cumpla con las necesidades del programador, además que permitan entrelazar librerías y bibliotecas externas, y configuración con otro tipo de software de programación.
- Manejar un lenguaje de programación único, como en este caso el lenguaje C++ y desarrollar el ejecutable para la interfaz de manera fácil sin compilación.
- Creación de un proyecto en el cual se pueda desarrollar trabajos a futuro.

5.2.1 Blender

Blender es un programa informático multi-plataforma, dedicado especialmente al modelado, iluminación, renderizado, animación y creación de gráficos tridimensionales [22]. Este programa se utilizó para desarrollar los actuadores en la interfaz, para este caso el área de reconocimiento (cubo u objeto inmóvil), y el objeto que simula la trayectoria y orientación (Objeto móvil - flecha).

5.2.2 Visual Studio

Como plataforma de programación se utiliza MICROSOFT® Visual Studio Ultimate 2010 [23]. Este software posee una colección completa de herramientas y servicios que permiten crear una gran variedad de aplicaciones; tanto para plataformas de Microsoft como para otras plataformas. Visual Studio conecta también todos los proyectos, equipos y partes interesadas.

5.2.3 QT

Para el desarrollo de la interfaz gráfica de usuario se utiliza Qt Project, el cual contiene una biblioteca multiplataforma, ampliamente usada para desarrollar aplicaciones con interfaz gráfica. La principal ventaja de esta plataforma es que es un software libre y de código abierto, bajo el lenguaje de programación C++.

Adicionalmente puede ser utilizado en varios otros lenguajes de programación.

Cabe aclarar que todas las herramientas de Qt son desarrolladas bajo el concepto *open-source* [24].

5.2.4 LIBRERIAS VTK

VTK (*visualization Toolkit*) es un software disponible de código abierto, el cual consiste en una biblioteca de clases C++. Este software soporta una amplia variedad de algoritmos de visualización incluyendo: escalar, vector, tensor, textura y métodos volumétricos. VTK tiene un amplio marco de visualización de información, además cuenta con un conjunto de *widgets* de interacción 3D, soporta el procesamiento en paralelo, y se integra con varias bases de datos de herramientas GUI como QT y VTK.

5.2.5 Librerías VCOLLIDE

V_Collide es una biblioteca de detección de colisiones para entornos de gran tamaño, está diseñado para funcionar con un gran número de objetos poligonales. Utiliza también una arquitectura de detección de colisiones en 3 etapas, explicadas en [25].

Los pasos para la integración de estas librerías a Visual Studio se explican en los anexos, en la sección A5.

5.2.6 CMAKE

CMake se utiliza para controlar el proceso de compilación del software; utilizando una plataforma simple y un compilador de archivo. Este programa permite configurar el enlace entre Qt y VTK generando los archivos necesarios para poder obtener una interfaz de usuario.

5.3 CONSTRUCCIÓN DE LA INTERFAZ DE USUARIO.

En la construcción de la interfaz primero se deben instalar todos los programas anteriores con sus respectivas bibliotecas y corregir los errores de instalación. Los errores de instalación se deben a que los programas no se ejecutan en su respectivo orden. Su correcta instalación se puede observar en el Anexo A.

De acuerdo al modo de funcionamiento de la HAPTIC 6DOF, se realiza un esquema de desarrollo de todo el sistema (Ver Figura 5. 2). En este esquema se observa el orden de funcionamiento de la interfaz y su modo de operación.

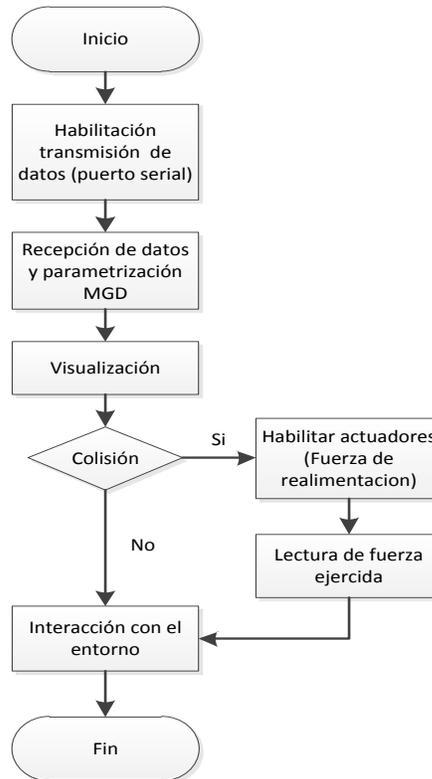


Figura 5. 2 Diagrama de flujo del funcionamiento.

Para el desarrollo de los objetos y actuadores de la interfaz se utiliza Blender. En la Figura 5. 3, se observa el desarrollo del objeto móvil, en este caso se define inicialmente una esfera. Además Blender permite diseñar este objeto con el sistema métrico (MKS), lo cual facilita el desarrollo de las demás piezas en relación con el entorno virtual, aparte de posicionar los objetos para que se adecuen bien con la interfaz, es decir que permite la interacción adecuada entre los objetos.

Una vez desarrollados los gráficos 3D se guardan en formato obj, el cual es ideal para el renderizado y visión del objeto en la interfaz (Ver Figura 5.3).

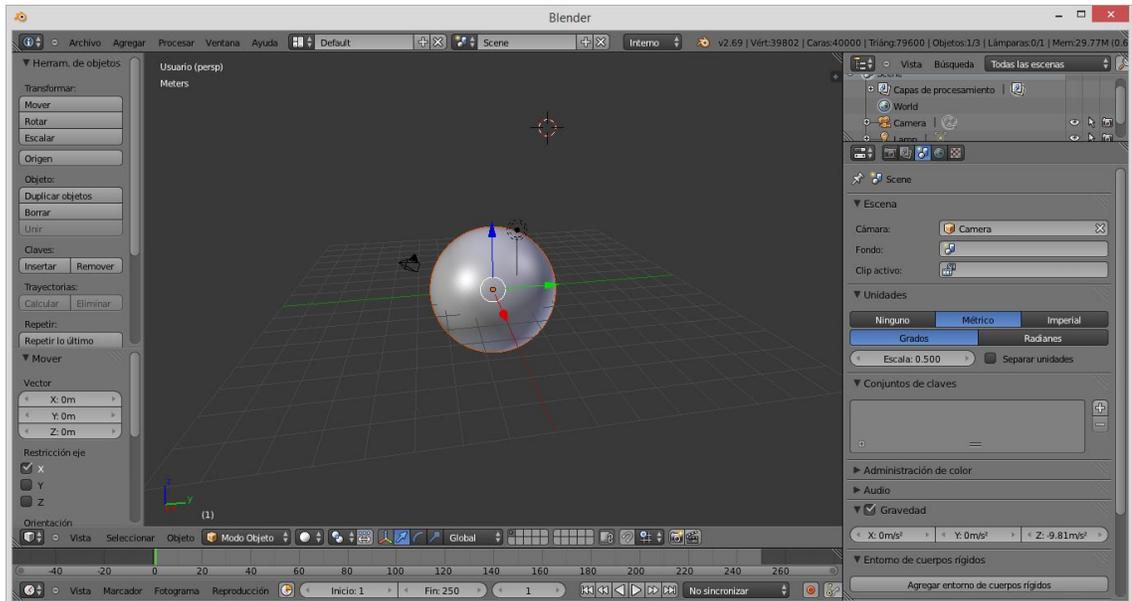


Figura 5. 3 Desarrollo de una esfera en Blender.

En el proyecto se agregan los objetos tridimensionales en una posición x y z de la interfaz de usuario, lo cual permitirá ver la posición y la orientación respectiva de acuerdo al funcionamiento de la HAPTIC 6DOF (Ver Figura 5.4).

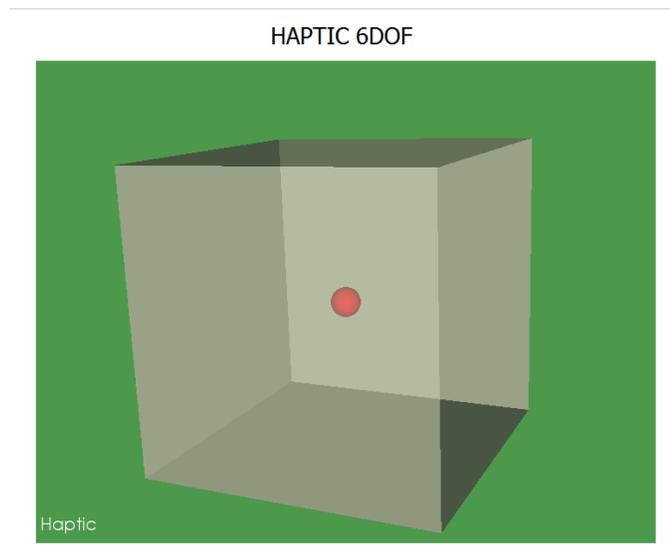


Figura 5. 4 Integración de los objetos 3D.

El desarrollo de la interfaz se inicia con el programa QT Creator, el cual permite generar una ventana, que incluye la opción de habilitar la comunicación entre ARDUINO y Visual Studio, además de crear una sub-ventana para el renderizado de los objetos 3D y los indicadores para la identificación del tipo de realimentación que se obtiene a través de la HAPTIC 6DOF (Ver Figura 5.5).

Previamente estos objetos son modelados en BLENDER y exportados con el formato *wavefront* (.obj), el cual es una extensión que permite guardar las características de los objetos en 3D.

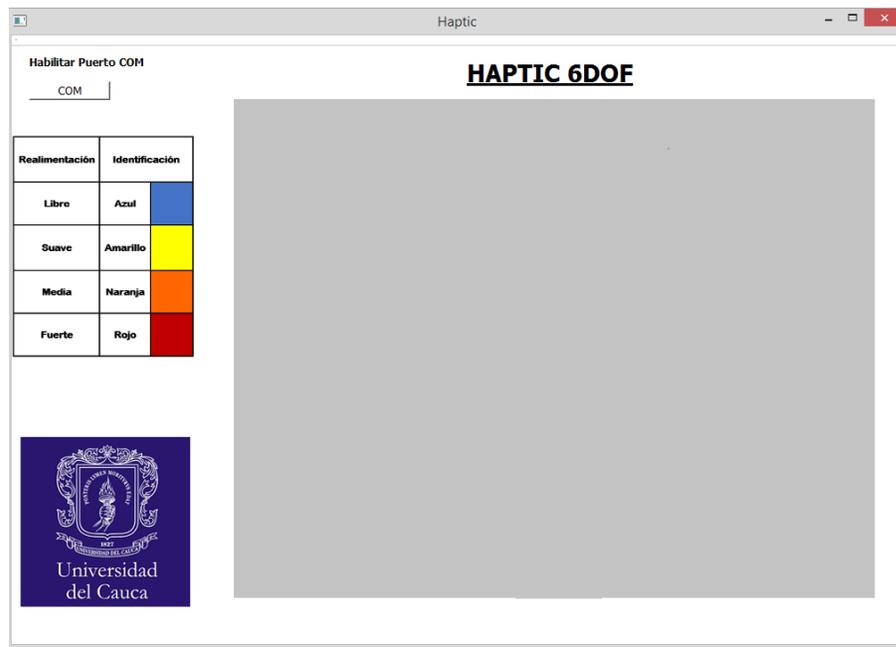


Figura 5. 5 Desarrollo de la Interfaz en Qt.

Una vez desarrollada la interfaz de usuario, se procede a generar el proyecto a través del generador de código CMake. Este programa generará el respectivo proyecto de la interfaz, para poder editarlo por medio de la plataforma Visual Studio, la cual ya tiene acceso a las librerías de VTK y VCollide.

La proyección de los objetos 3D a través de la interfaz se realiza por medio de VTK. Para la integración de los objetos 3D se crea una clase llamada Objetos, la cual toma la lectura de los objetos creados, proporcionando características necesarias para generar el renderizado.

En este proceso se toma la librería de vtkObjReader, la cual lee los objetos creados a través de Blender.

Para el filtro de la imagen se utiliza la librería vtkPolyData, la cual representa una estructura geométrica que consiste en vértices, líneas, polígonos y líneas de triángulos, adicionalmente esta librería permite la manipulación en el proceso de los datos obtenidos.

Siguiendo este proceso se utiliza vtkPolyDataMapper la cual asigna datos poligonales, realizando un mapeo para la prestación de gráficos de hardware/software.

A continuación se muestra parte del código anteriormente mencionado y la respectiva tabla de explicación por cada línea de código.

```
obj[j] = vtkOBJReader::New(); // 1
polyData[j]= vtkPolyData::New(); // 2
mapper[j] = vtkPolyDataMapper::New(); // 3
actor[j] = vtkActor::New(); // 4
obj[j]->SetFileName(file[j]); // 5
obj[j]->Update(); // 6
polyData[j]=obj[j]->GetOutput(); // 7
mapper[j]->SetInput(polyData[j]); // 8
actor[j]->SetMapper(mapper[j]); // 9
matriz[j]=vtkMatrix4x4::New(); // 10
actor[j]->SetUserMatrix(matriz[j]); // 11
```

Línea de código	Explicación
1	Se crea un objeto de la clase “vtkOBJReader” y se asigna a “obj[j]”, para la lectura de objetos 3D.
2	Se crea un objeto de la clase “vtkPolyData” y se asigna a polyData[j], para la estructura geométrica del objeto 3D.
3	Se crea un objeto de la clase “vtkPolyDataMapper” y se asigna a “mapper[j]”, para el mapeo del objeto 3D.
4	Se crea un objeto de la clase “vtkActor” y se asigna a “actor[j]”, para la representación de un objeto 3D en una escena renderizada.
5	Asigna el objeto 3D identificado a la instancia “obj[j]”
6	Actualiza el objeto 3D
7	Consigue la salida de la lectura “obj[j]”
8	Especificación de los datos para mapear
9	Este método se utiliza para conectar un actor a la fuente de información de la visualización.
10	Se crea un objeto de la clase “vtkMatrix4x4” y se asigna a “matriz[j]”, para representar y manipular matrices de transformación 4x4
11	Agrega una transformación adicional para su propio uso. Esta transformación se concatena con la transformación del actor

interno, que se crea de forma implícita a través del uso de SetPosition () , SetOrigin () y SetOrientation () (aplicaciones de la posición , origen y orientación de los actores móviles)

Tabla 5. 1 Explicación de líneas de código.

En el proceso de la posición y orientación se crea una clase Eventos, la cual permite una vez leído los objetos creados, relacionarlos con los datos que se reciben de la comunicación serial.

En esta clase se incluye la librería vtkMatrix4x4 la cual permite representar y manipular matrices 4x4, con el propósito de diseñar coordenadas para los objetos 3D. En este caso se utiliza para la detección de colisiones reconociendo las coordenadas de las dimensiones del cubo en el espacio.

En la clase Eventos se incluye el MGD, el cual realiza la recepción de los datos y el proceso de generar las posiciones cartesianas a partir de los ángulos generados por los actuadores principales. Se logra así obtener la posición y orientación del actor móvil en la interfaz.

A continuación se muestra parte del código anteriormente mencionado y la respectiva tabla de explicación por cada línea de código.

```

Art_1 = TextBuff[1]; // 1
Art_2 = TextBuff[2]; // 2
Art_3 = TextBuff[3]; // 3
Art_4 = TextBuff[4]; // 4
Art_5 = TextBuff[5]; // 5
Art_6 = TextBuff[6]; // 6

//----- Adecuación coordenadas -----//

double t1 = double(Art_1); // 7
double t2 = double(Art_2); // 8
double t3 = double(Art_3); // 9

double t4 = double(Art_4); // 10
double t5 = double(Art_5); // 11
double t6 = double(Art_6); // 12

//----- Posición cartesiana -----//

double Px, Py, Pz; // 13
double sx, sy, sz, nx, ny, nz, ax, ay, az; // 14

//----- MGD Posición -----//

Px = (-cos(t1)*(sin(t2 + t3)*R4 - cos(t2)*D3)); // 15
Py = (-sin(t1)*(sin(t2 + t3)*R4 - cos(t2)*D3)); // 16
Pz = (cos(t2 + t3)*R4 + sin(t2)*D3); // 17

```

Línea de código	Descripción
1-6	Recepción de los datos desde la tarjeta Arduino de las articulaciones 1, 2, 3, 4, 5 y 6 respectivamente.
7-12	Conversión y asignación de los datos de recepción de tipo "char" a tipo "double" para las operaciones matemáticas e impresión de los resultados.
13	Definición de las variables de Posición
14	Definición de las variables de la matriz de orientación.
15-17	Aplicación y asignación del Modelo Geométrico directo para la posición del efector final

Tabla 5. 2 Explicación de líneas de código

Una vez se ha realizado todo el proceso del desarrollo de la interfaz de usuario, inclusión de librerías y creación objetos actores, se obtiene como resultado la siguiente interfaz (Ver Figura 5.6)

Inicialmente se utilizó como actor móvil una esfera para probar las articulaciones de posicionamiento, posteriormente con el fin de realizar pruebas con las articulaciones de orientación se creó un actor móvil similar a una flecha; esto con el objetivo de diferenciar sobre que eje rota dicho actor.

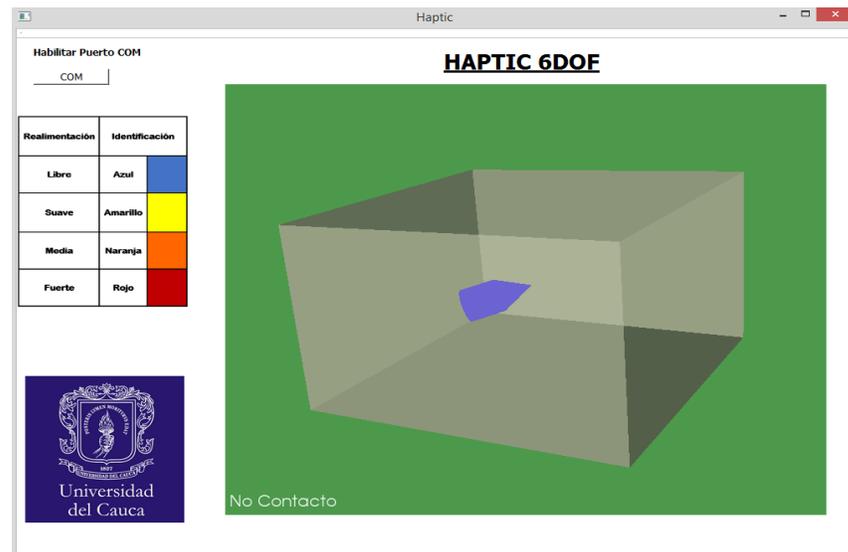


Figura 5. 6 Interfaz de usuario HAPTIC 6DOF.

5.3.1 Comunicación interfaz – arduino

Una vez creado el proyecto se empieza a desarrollar la comunicación serial Arduino-Visual Studio. Para este caso se obtiene un código de comunicación, el cual permite establecer la comunicación simultánea (transmisión - recepción) entre el entorno y la tarjeta Arduino.

La interfaz de usuario recibirá 6 datos de la tarjeta Arduino, los cuales corresponden a las 6 articulaciones respectivamente, generando a través del MGD las posiciones cartesianas y la orientación del objeto móvil.

En la adquisición de datos del entorno se desarrolla una estructuración de código que permite configurar el puerto; además de configurar el tamaño del buffer de información y los datos a enviar respectivamente [26]. El código serial de comunicación se define con base en la siguiente tabla.

Función	Descripción
Tserial ()	Creación de objetos
~ Tserial ()	Destrucción de objetos
int connect (char *port_arg, int rate_arg, serial_parity parity_arg)	Configuración del puerto serie y el inicio de la conexión. <ul style="list-style-type: none"> - Port_arg: "COM1", "COM2",... - Rate_arg: 9600, 19600,... - Parity_arg: spNONE, spODD, spEVEN Devuelve 0 si ha ocurrido algún error.
void sendChar (char c)	Envía un tipo char en la línea serial.
void sendArray (char *buffer, int len)	Envía una matriz de tipo char en la línea serial hasta que se envía una trama.
char getChar (void)	Devuelve el siguiente carácter del búfer de entrada. Se bloquea si no hay datos disponibles.
int getArray (char *buffer, int len)	Devuelve una matriz del búfer de entrada. Se bloquea hasta que están disponibles los datos (algunos), el parámetro devuelto indica cuantos bytes se han leído.
int getNbrOfBytes (void)	Devuelve el número de bytes de espera en el búfer de entrada.
void disconnect (void)	Desconecta el puerto serial.

Tabla 5. 3 Configuración Tserial (comunicación serial) [6].

El tipo de datos que recibirá el entorno 3D serán valores en grados los cuales son las posiciones de las articulaciones. Para determinar el espacio de trabajo adecuado en el entorno 3D es necesario establecer los límites de trabajo de HAPTIC 6DOF, efectuando una relación entre estos datos, como se muestra en la Tabla 5.4. desarrollo

Articulación	Angulo haptic min.	Angulo haptic max.	Angulo entorno virtual min. (cubo)	Angulo entorno virtual max. (cubo)
1	0	180	0	180
2	0	110	0	110
3	0	100	0	100
4	0	180	0	180
5	0	180	0	180
6	0	360	0	360

Tabla 5. 4 Espacios de trabajo interfaz – entorno 3D.

Una vez son leídos los datos por la tarjeta Arduino, la plataforma de Visual Studio procesa estos datos a través del MGD, leyendo las posiciones articulares de los servos, para posteriormente generar las posiciones cartesianas y de orientación del efector final.

En la transmisión de datos Entorno – ARDUINO se maneja la librería de colisiones V_collide, la cual permite por medio de objetos poligonales presenciar colisiones entre objetos. En este proyecto se utiliza un cubo como espacio de trabajo; el cual es un objeto inmóvil y genera una identificación de cada una de las paredes, permitiendo saber su orientación con respecto al objeto móvil (esfera). Esta identificación del punto de colisión se envía hacia el puerto serial; indicando sobre qué articulación se produce el segundo evento (Capítulo 4.4).

A continuación se muestra parte de los comandos utilizados para tal fin.

```
//RECEPCION:
char TextBuff[128]; // tamaño de la trama de recepción.
this->rx_arduino(); // llamado de la clase recepción.
com->getArray(TextBuff,10); // recepción de trama de 10 datos.

//TRANSMISION:
void Eventos::tx_arduino() // invoca la transmisión
int señaltx; // definición del tipo de señal que se presenta en la colisión.
```

```
señaltx = colisiones->GetIdCeldaContacto(); // asignación de la id de la celda.  
com->sendChar(señaltx); // transmite el punto de colisión.
```

5.3.2 Detección de colisiones

Este algoritmo se utiliza en el entorno de la interfaz para provocar eventos entre el objeto móvil (esfera) y un objeto inmóvil (cubo).

Para este proceso se representa un conjunto de datos de los objetos creados y se genera triángulos sobre el objeto 3D. Al ser posicionados en el espacio se identifican estos triángulos y se genera una ID la cual define el punto del área en donde se produce la colisión.

Una vez se presenta el evento de colisión se genera un dato de confirmación de dicho evento, que es enviado por medio del puerto serial a la tarjeta Arduino.

A continuación se muestra la línea de código, la cual asigna una id de la ubicación del punto de colisión, al objeto creado señaltx.

```
//LECTURA ID:  
señaltx = colisiones->GetIdCeldaContacto();
```

5.3.3 Programación Arduino

En la tarjeta ARDUINO MEGA 2560 se implementa la programación para la lectura de cada una de las articulaciones, además de reconocer los actuadores correspondientes a cada una de ellas.

De acuerdo a las primeras 3 articulaciones que en este caso controlan la posición del objeto móvil de la interfaz, se cuenta con servomotores digitales que tienen como características (manual de operación) lectura serial de la posición en la que se encuentran además de representar las posiciones en forma de ángulos (-160° a 160°) o tensiones entre 0 y 5 voltios a un número entero entre 0 y 1023. En este caso debido al funcionamiento del MGD se emplea la lectura de la posición en grados para facilitar la programación en la plataforma de Visual Studio.

Para la aplicación y uso de estas características se debe emplear las librerías de Herkulex, las cuales se descargan en la página de este producto [20].

Las librerías de Herkulex, se insertan en las librerías de instalación de Arduino para su aplicación (Ver Anexo B).

Una característica de estos servomotores es que se puede habilitar el torque de movimiento, es decir se manejará un torque libre para que la HAPTIC 6DOF se manipule libremente en el espacio, y se aplicará un troque para detener el movimiento de una articulación en caso de algún evento, en nuestro caso una colisión.

En la configuración del funcionamiento de cada servomotor Herkulex se utiliza una lista de comandos (Ver Tabla 5.5), la cual permite manipular este mecanismo de una manera eficiente y ordenada. Estos comandos son de fácil utilización y aplicación.

COMANDO	FUNCION
begin(long baud, int rx, int tx)	Iniciar la comunicación entre ARDUINO MEGA y HERKULEX.
beginSerial1(long baud)	Iniciar la comunicación entre ARDUINO MEGA y HERKULEX mediante el puerto serie 1.
initialize()	Inicializar todos los motores.
stat(int servoid)	Pregunte por el estado del motor para controlar los errores.
set_ID(int ID_Old, int ID_New)	Cambiar el ID del motor.
torqueON(int servoid)	Establecer el par del motor en ON.
torqueOFF(int servoid)	Establecer el par del motor en OFF.
moveOne(int servoid, int Goal, int pTime, int iLed)	Mover un motor a una posición absoluta.
moveOneAngle(int servoid, float angle, int pTime, int iLed)	Mover un motor a un ángulo.
getAngle(int servoid)	Obtener la posición en grados.
reboot(int servoid)	Reiniciar el motor.
setLed(int servoid, int valueLed)	Establecer el color del LED.

Tabla 5. 5 Comandos de los servomotores Herkulex en ARDUINO.

En la conexión con la tarjeta ARDUINO se escoge el módulo de conexión de comunicación y se escoge un lazo de transmisión y recepción y una tierra común con ARDUINO para su funcionamiento (Ver Anexo B).

La alimentación de los servomotores se hace aparte ya que consume un voltaje de 7 - 7.4 voltios

Tanto para los actuadores secundarios como para los sensores, las librerías vienen integradas en la plataforma de ARDUINO para su lectura y control.

6 RESULTADOS.

En este capítulo se diferenciarán los resultados obtenidos. Primero explicando los logros que se consiguieron cuando la interfaz se mueve en la zona libre de realimentación de fuerzas o primer evento. Luego se explicaran los resultados logrados cuando la interfaz se encuentra con un punto de colisión o segundo evento y la diferenciación entre los tipos de realimentación que se implementaron. Por último se mostrará cómo actúa la orientación del objeto móvil dentro del entorno 3D.

6.1 RESULTADOS EN LA ZONA LIBRE.

Para obtener un adecuado funcionamiento de la interfaz dentro de la zona libre, se debe garantizar primero que las placas de acrílico y aluminio se encuentren ancladas, logrando que los datos obtenidos de las articulaciones sean entregados por los actuadores principales y puedan ser ingresados al modelo geométrico directo. El segundo requerimiento consiste en que las placas se anclen correctamente en el momento que el usuario haga el reingreso a la zona libre, después de haberse presentado un evento de colisión.

En el caso de que no se presente ninguna colisión, el usuario podrá ver el objeto móvil de color azul. Esto con el fin de diferenciar los dos tipos de eventos que se pueden generar.

Para mostrar cómo funciona la interfaz frente a este evento, se realizaron diferentes movimientos en los tres ejes.

Primero para probar los movimientos en el eje x, se realizó una línea. En las siguientes figuras se muestran dos de los puntos en el entorno 3D (Ver Figura 6.1) y en la Figura 6.2 se muestra la posición de HAPTIC 6 DOF.

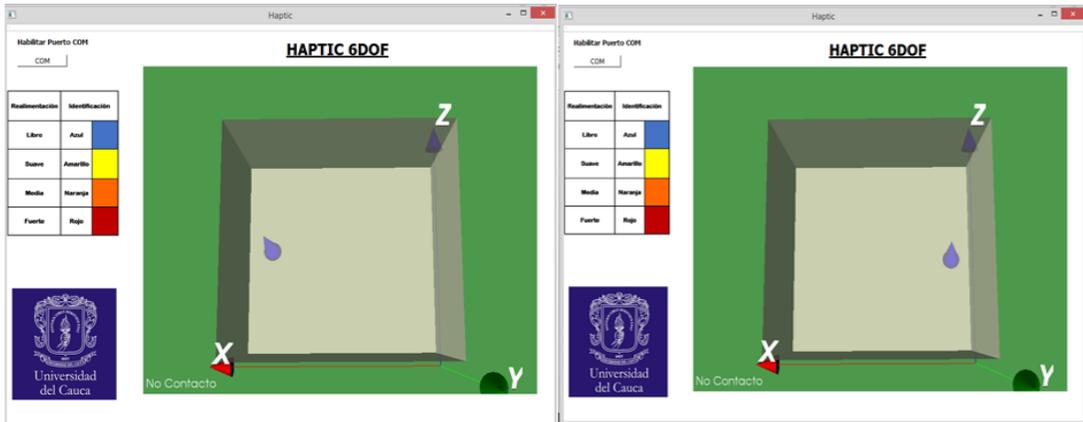


Figura 6. 1 Dos puntos de la línea en el entorno 3D – eje x.



Figura 6. 2 Dos puntos de la línea en HAPTIC 6DOF – eje x

Para el eje y también se realizó una línea. En las siguientes figuras se muestran dos de los puntos en el entorno 3D (Ver Figura 6.3) y en la Figura 6.4 se muestra la posición de HAPTIC 6 DOF.

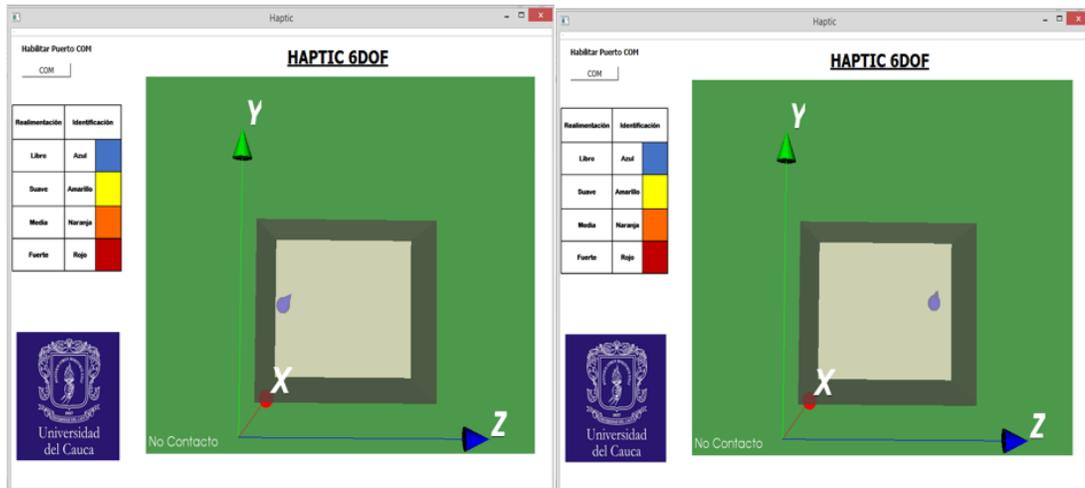


Figura 6. 3 Dos puntos de la línea en el entorno 3D – eje Y.



Figura 6. 4 Dos puntos de la línea en HAPTIC 6DOF – eje y

Para el eje z también se realizó una línea. En las siguientes figuras se muestran dos de los puntos en el entorno 3D (Ver Figura 6. 5) y en la Figura 6.6 se muestra la posición de HAPTIC 6 DOF.

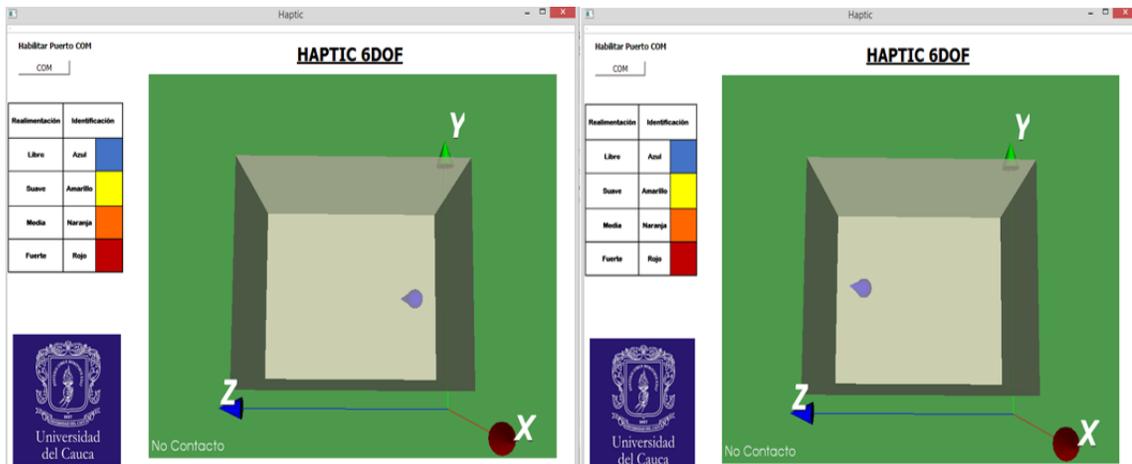


Figura 6. 5 Dos puntos de la línea del entorno 3D – eje z.



Figura 6. 6 Dos puntos de la línea en HAPTIC 6DOF – eje y

6.2 RESULTADOS ZONA DE REALIMENTACIÓN.

En la zona de realimentación se realizan varios eventos mecánicos, los cuales permiten tener una aproximación de la realimentación de fuerzas. En primer lugar cuando ocurre una colisión las placas de aluminio y de acrílico se desanclan, gracias al actuador secundario. Posteriormente dependiendo del valor del potenciómetro se genera una fuerza de realimentación que es realizada por el actuador principal en conjunto con los resortes. Como consecuencia de estos eventos mecánicos tenemos una aproximación de la deformación que se hace a un objeto en un ambiente real.

Para que el usuario que interactúa con el entorno, diferencie visualmente los tipos de realimentación se utilizaron tres colores; estos se muestran dependiendo de la realimentación que se esté efectuando. Es necesario aclarar que las operaciones que se realizan para sentir estos tres tipos de realimentación están explicadas en la sección 4.4 CONSTRUCCIÓN DEL SISTEMA DE REALIMENTACIÓN.

Para la primera realimentación (suave) el color que se predefinió para el objeto móvil fue el amarillo. A continuación se muestra una imagen de un evento de colisión realizado en el eje x.

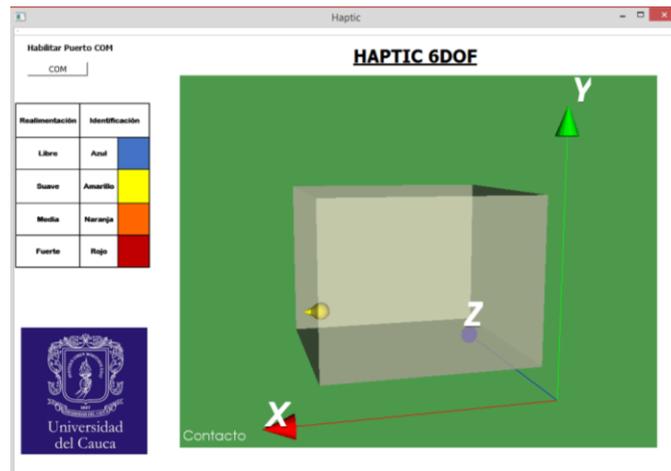


Figura 6. 7 Realimentación suave.

Para la segunda realimentación (media) se prefijo el color naranja para el objeto móvil. En la siguiente figura se muestra un evento de colisión sobre el eje y.

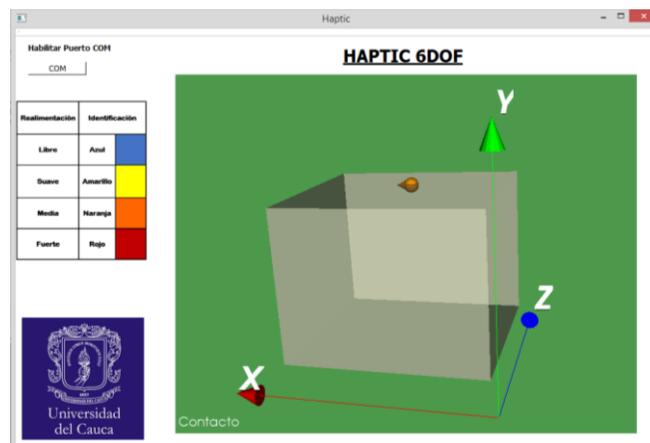


Figura 6. 8 Realimentación media.

Para la tercera realimentación (fuerte) se prefijo el color rojo para el objeto móvil. En la siguiente figura se muestra un evento de colisión sobre el eje z.

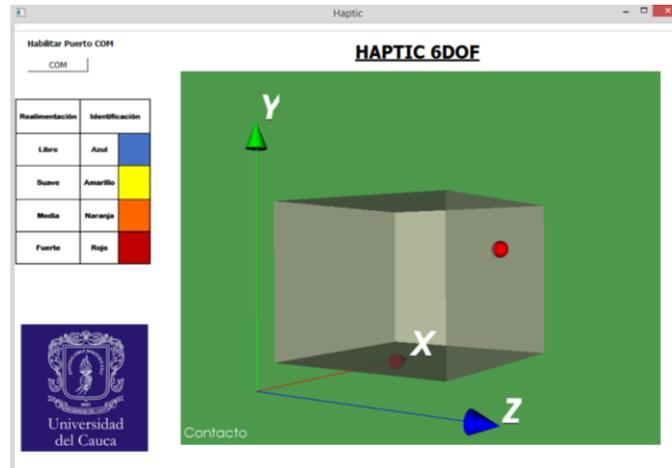


Figura 6. 9 Realimentación fuerte

6.3 RESULTADOS PARA ORIENTACIÓN

Para mostrar correctamente la orientación del objeto móvil; que en este caso es un objeto similar a un cono. De esta manera se puede diferenciar sobre que eje se está realizando la rotación.

En la Figura 6.10 se muestran dos puntos en el eje x del entorno 3D y en la Figura 6.11 sus respectivos puntos reales en HAPTIC 6DOF.

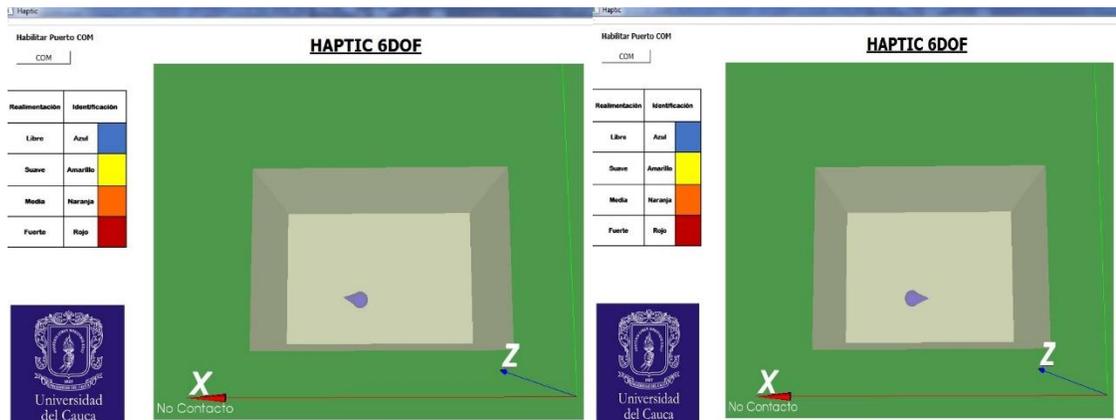


Figura 6. 10 Puntos de rotación en el eje X – entorno 3D.



Figura 6. 11 Puntos de orientación en el eje x – HAPTIC 6DOF

En la Figura 6.12 se muestran dos puntos en el eje y del entono 3D y en la Figura 6.13 sus respectivos puntos reales en HAPTIC 6DOF.

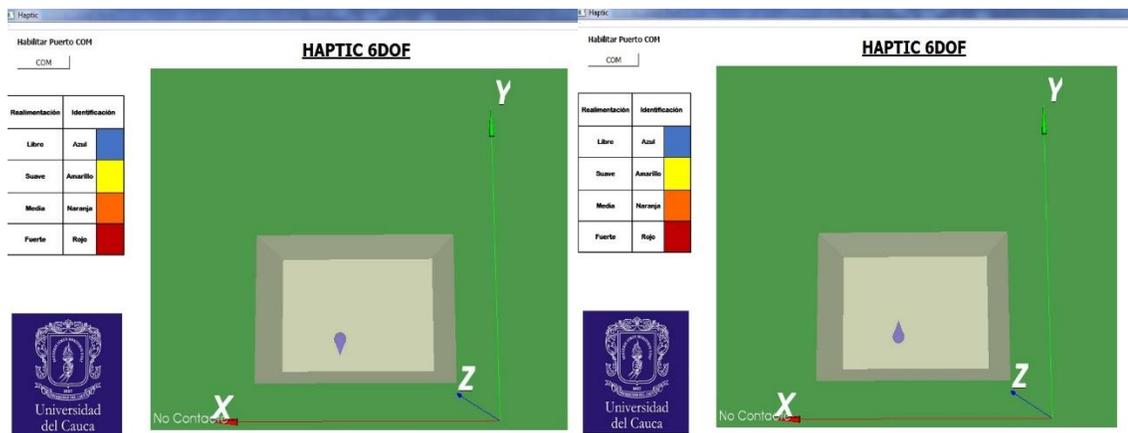


Figura 6. 12 Puntos de rotación en el eje Y.



Figura 6. 13 Puntos de orientación en el eje y – HAPTIC 6DOF.

En la Figura 6.14 se muestran dos puntos en el eje Z del entono 3D y en la Figura 6.15 sus respectivos puntos reales en HAPTIC 6DOF.

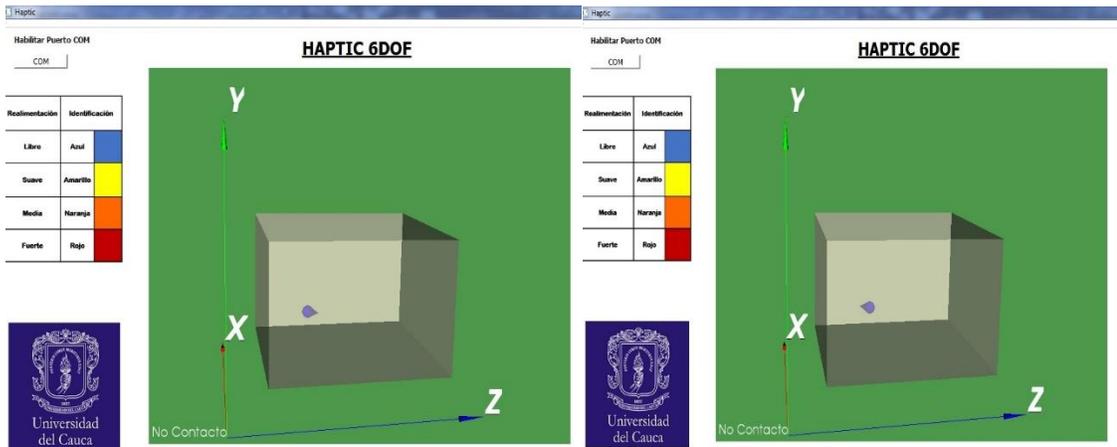


Figura 6. 14 Puntos de rotación en el eje Z.



Figura 6. 15 Puntos de orientación en el eje z – HAPTIC 6DOF

7 CONCLUSIONES Y TRABAJOS FUTUROS

HAPTIC 6 DOF es el segundo prototipo de interfaz háptica realizado en la Universidad del Cauca, que genera realimentación de fuerzas.

Este proyecto aporta mejoras en cuanto a la realimentación de fuerzas ya que adiciona 3 tipos de realimentación, que simulan sentir tres tipos de fuerzas distintas, a diferencia del primer prototipo que solo ejerce el frenado completo al tocar una de las paredes del entorno de trabajo. Adicionalmente el proyecto sirve como guía para que se sigan creando interfaces hápticas de bajo costo y que puedan ser utilizadas por estudiantes y docentes.

Una de las formas para optimizar la construcción de HAPTIC 6 DOF fue el haber utilizado la herramienta CAD para realizar el diseño previo de la interfaz, ayudando a disminuir el tiempo y costo de fabricación del prototipo, ya que primero permite tener una idea general del dimensionamiento total de la interfaz, logrando de esta manera corregir errores que se puedan presentar al acoplar las piezas que forman la estructura. Adicionalmente, este software entrega las masas y las distancias a los centros de masa de cada articulación, las cuales son usadas para hallar los valores del primer momento de inercia; mientras que el segundo momento de inercia es obtenido directamente del software. Finalmente estos valores son ingresados en los modelos dinámicos.

Por otra parte la simulación se basa en los modelos geométricos y dinámicos, utilizando la estructura de un controlador CTC, una consigna circular y unas ganancias convenientes, con el fin de obtener los pares o fuerzas que van a los servomotores de cada articulación, permitiendo conocer el tipo de actuador necesario para la construcción del prototipo.

Gracias al previo diseño y simulación de la interfaz se lograron establecer los diferentes componentes y materiales necesarios para un adecuado funcionamiento de la interfaz. Sin embargo para la etapa de construcción se hace necesario contar con una herramienta especializada debido a que el ensamblaje de la interfaz requiere exactitud para su buen funcionamiento.

Con algunas pruebas de funcionamiento y trabajos previos relacionados con Robótica se establecieron los componentes electrónicos aptos para la comunicación y alimentación de la interfaz, siendo la comunicación una de las tareas que más requirió trabajo en el proyecto, dado que se hacía necesario primero proporcionar una correcta transmisión y recepción de los datos de HAPTIC 6DOF, y segundo buscar una tarjeta de adquisición de datos que tuviera una plataforma de programación sencilla y flexible.

Se logró concluir que los programas y librerías utilizados en la implementación del entorno 3D fueron de sencilla vinculación, con lo cual se logró obtener un entorno que se aproxima al verdadero realismo visual.

En la etapa final del proyecto se ensambló completamente la interfaz culminando satisfactoriamente con la etapa de construcción. Finalmente se realizaron múltiples pruebas de comunicación entre la interfaz háptica y el entorno de trabajo tridimensional llegando así a la culminación del proyecto con resultados satisfactorios; pero abierto para mejoras futuras.

7.1 TRABAJOS FUTUROS

- Implementar un sistema que permita diferenciar entre las fuerzas que se ejercen en función de la profundidad de contacto del material utilizado y las fuerzas que deben realizar los resortes, más específicamente vincular las matriz jacobiana del resorte con el fin de obtener con más precisión las posiciones angulares de los actuadores principales en relación con los tipos de realimentación que se estén efectuando, en el siguiente diagrama de bloques se explica con más detalle el sistema que se propone crear para mejorar la precisión de la interfaz.

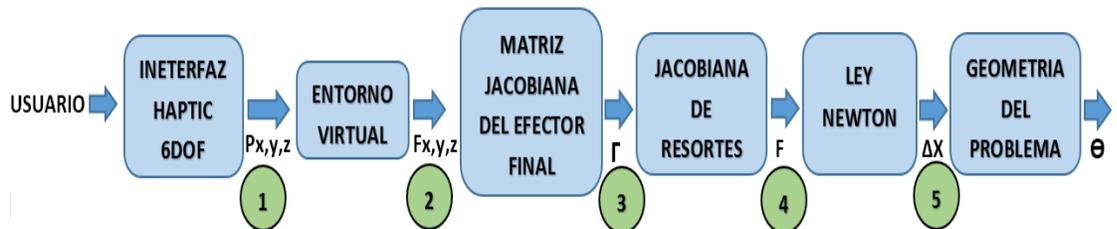


Figura 7. 1 Diagrama de bloques del sistema propuesto.

A continuación se explica con más detalle el sistema, de acuerdo con los numerales que se muestran en el diagrama.

Inicialmente tenemos el usuario como una fuente de información para la interfaz.

1. la interfaz háptica entrega posiciones cartesianas x,y,z del efector final, al entorno virtual.

2. el entorno virtual, calcula las fuerzas en función de la profundidad de contacto.

3. con la matriz jacobiana del efector final se relacionan las fuerzas obtenidas y se calculan los torques articulares. (Esto se realiza ya que no se poseen actuadores de fuerza).

4. con la matriz jacobiana de los resortes y los torques articulares se pueden hallar las fuerzas en los resortes.

Es necesario aclarar que la jacobiana de resortes depende de los puntos donde los resortes se unen a la interfaz, para obtener la velocidad lineal de los puntos de contacto de los resortes en función de las velocidades articulares.

5. finalmente, una vez se tienen las fuerzas en los resortes y con la ley de newton se pueden hallar las elongaciones en los resortes, las cuales pueden ser relacionadas con las posiciones angulares de los actuadores principales gracias a la geometría del dispositivo.

- Construir una interfaz de usuario en Qt u otra aplicación adecuada, que permita configurar y representar la realidad virtual, provocando eventos tales como el cambio de los objetos 3D, además de representar información necesaria (posición actual en grados) de las articulaciones y el estado de los dispositivos electrónicos como motores y sensores (torque).

Rediseñar la realidad virtual del entorno 3D por medio de las bibliotecas de VTK, las cuales incluyen más opciones para mejorar la calidad de visualización, una adecuada renderización de los objetos actores (cubo y puntero) y configuración de la posición de la cámara de avistamiento, que es muy importante en el seguimiento de las trayectorias de los objetos móviles en el área de trabajo.

Agregar objetos 3D más representativos con respecto al material simulado, (lona, hierro, polietileno expandido, icopor, etc.) y representarlos por medio de la interfaz de usuario para la simulación de la fuerza de realimentación y así obtener una aproximación en un ambiente real.

Reestructurar el control de realimentación en la aplicación Arduino u otra plataforma de programación, para el cambio de material simulado y configurarlo para ser controlado desde la interfaz de usuario.

- Integrar y desarrollar una clase de deformación virtual para cada material relacionándolo apropiadamente con el ambiente real y que sea representado en la interfaz.

Utilizando las librerías apropiadas de VTK u otras bibliotecas relacionadas con el entorno C++, el actor u objeto 3D al cual se le va aplicar la clase deformación, se debe estructurar de acuerdo al tipo de fuerza que se le aplicará y la textura del material. Esto debe ser reflejado tanto en el dispositivo háptico como en la interfaz de usuario 3D.

- Cambiar los actuadores y sensores secundarios con el fin de mejorar la efectividad del dispositivo, tanto para el anclaje y des anclaje de las placas frente a los dos tipos de eventos que se presentan en la interfaz. También es necesario mejorar el sistema de censado ya que se necesita ser más preciso con los valores que se obtienen cuando el usuario ejerce una fuerza en el momento que se presenta un evento de colisión.

8 Bibliografía

- [1] J. J. Gil, «Control de Dispositivos Físicos de Gran Espacio de Trabajo para la Interacción Táctil con Entornos Virtuales,» Tesis de Grado, Universidad de Navarra, San Sebastian, España, 2003.
- [2] M. L. Pinto, «Análisis e Implementación de una Interfaz Háptica en Entornos Virtuales,» Tesis de Grado, Universidad Nacional de Colombia, Bogota, Colombia, 2009.
- [3] Novint Technologies Inc, «Gaming & Consumer Products NOVINT,» 2012. [En línea]. Available: <http://www.novint.com/index.php/novint/company>. [Último acceso: 14 Agosto 2013].
- [4] VertMarkets Inc, «Hospital Network,» [En línea]. Available: <http://www.hospitalnetwork.com>. [Último acceso: 14 Agosto 2013].
- [5] CyberGlove Systems Inc, «Cyber Glove Systems,» [En línea]. Available: <http://www.cyberglovesystems.com/products/cyberforce/overview>. [Último acceso: 14 Agosto 2013].
- [6] Toyota Motor Corporation, «Human Support Robot (HSR) Prototype,» [En línea]. Available: <http://www2.toyota.co.jp/en/news/12/09/0921.html>. [Último acceso: 14 Agosto 2013].
- [7] T. M. Piamba , «Sistema de Interfaz Háptica para el Control de Posición en un Espacio Tridimensional Virtual,» Tesis Grado, Universidad del Cauca, Popayan, Colombia, 2011.
- [8] J. San Marín, «Aportaciones al Diseño Mecánico de los Entrenadores basados en Realidad Virtual,» Tesis Doctoral, Universidad Rey Juan Carlos, Madrid, España, 2007.
- [9] Geomagic Sensable, «ClayTools Geomagic,» [En línea]. Available: <http://geomagic.com/en/products/claytools/overview>. [Último acceso: 20 Agosto 2013].
- [10] Université Laval, «SHADE (A Spherical 3-DOF Haptic Device),» [En línea]. Available: <http://robot.gmc.ulaval.ca/index.html>. [Último acceso: 20 Agosto 2013].
- [11] HAPTION S.A, «6D Virtuouse™,» [En línea]. Available: <http://www.haption.com/site/index.php/en/>. [Último acceso: 20 Agosto 2013].
- [12] Novint Technologies Inc, «Novint Falcon 3D,» [En línea]. Available: <http://home.novint.com/index.php/products/novintfalcon>. [Último acceso: 20 Agosto 2013].
- [13] VRLOGIC GmbH, «Immersion Cyberforce®,» [En línea]. Available: <http://www.vrlogic.com/html/immersion/cyberforce.html>. [Último acceso: 20 Agosto 2013].

- [14] MPB Technologies Inc, «7 DOF Haptic Interface,» [En línea]. Available: http://www.mpb-technologies.ca/mpbt/mpbt_web_2009/_en/7dof/. [Último acceso: 20 Agosto 2013].
- [15] Organizacion de Estados Iberoamericanos, «Ciencia con Tacto,» [En línea]. Available: http://www.oei.es/divulgacioncientifica/reportajes_414.htm. [Último acceso: 20 Agosto 2013].
- [16] W. Khalil y E. Dombre, «Modeling, Identification and Control of Robots,» *Kogan Page Science*, 2002.
- [17] O. A. Vivas , «Diseño y Control de Robots Industriales», Buenos Aires: Elaleph, 2010.
- [18] O. Vivas y S. Salinas, «Modelado, Simulación y Control del Robot para cirugía Laparoscópica "LAPBOT",» *INGENIARE*, vol. 17, pp. 324-325, 2009
- [19] SGBotic Pte. Ltd, «HerkuleX inteligente Servo,» 2014. [En línea]. Available: http://www.sgbotic.com/index.php?dispatch=categories.view&category_id=153. [Último acceso: 20 febrero 2014].
- [20] Arduino®, «Language Reference,» 2014. [En línea]. Available: <http://arduino.cc/en/Reference/HomePage>.
- [21] wikipedia.org, «Interfaz de usuario,» [En línea]. Available: http://es.wikipedia.org/wiki/Interfaz_de_usuario. [Último acceso: 11 febrero 2014].
- [22] Blender™, «Blender Website,» [En línea]. Available: <http://www.blender.org/about/website/>. [Último acceso: 11 febrero 2014].
- [23] Microsoft Corporation, «Visual Studio,» [En línea]. Available: <http://www.visualstudio.com/>. [Último acceso: 14 febrero 2014].
- [24] Qt Project © , «Qt Creator,» [En línea]. Available: http://qt-project.org/wiki/Category:Tools::QtCreator_Spanish. [Último acceso: 14 febrero 2014].
- [25] University of North Carolina, «Detección de Colisiones,» [En línea]. Available: <http://gamma.cs.unc.edu/V-COLLIDE/>. [Último acceso: 3 marzo 2014].
- [26] Tétraèdre Sarl Company, «API de software,» [En línea]. Available: <http://www.tetraedre.com/advanced/serial.php>. [Último acceso: 3 marzo 2014].
- [27] Carnegie Mellon University's Microdynamic Systems Laboratory, «The Maglev Haptic API Directory Structure,» [En línea]. Available: <http://butterflyhaptics.com/maglev-haptics/>. [Último acceso: 20 Agosto 2013].
- [28] Fundacion Wikipedia, «Wikipedia,» [En línea]. Available: http://es.wikipedia.org/wiki/Ley_de_elasticidad_de_Hooke. [Último acceso: 20 Junio 2014].