

Base para sistema de entrenamiento quirúrgico: robot LapBot



**María Cristina Chaparro Velasco
Ayxa Nataly Caicedo Rosero**

Director: PhD. Oscar Andrés Vivas Albán

**Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Electrónica, Instrumentación y Control
Línea de Robótica y Control
Popayán, Septiembre de 2013**

Base para sistema de entrenamiento quirúrgico: robot LapBot

**Documento Final de Trabajo de Grado para optar al título de
Ingeniero en Electrónica y Telecomunicaciones**

**María Cristina Chaparro Velasco
Ayxa Nataly Caicedo Rosero**

Director: PhD Oscar Andrés Vivas Albán

**Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Electrónica, Instrumentación y Control
Línea de Robótica y Control
Popayán, Septiembre de 2013**

Hoja de Aprobación

Director _____
Phd. Oscar Andrés Vivas

Jurado _____

Jurado _____

Fecha de sustentación _____

Agradecimientos

A Dios por sus infinitas bendiciones.

A nuestros padres por su amor, apoyo y fortaleza.

A nuestros amigos por brindarnos inspiración y alegrías.

A nuestro director de tesis Andrés Vivas por su apoyo tanto intelectual como personal, para la realización de este proyecto.

A los profesores por sus enseñanzas.

A nuestros compañeros de estudio por su apoyo y amistad.

A los evaluadores de este proyecto que con sus críticas constructivas fomentan el crecimiento y la mejora continua.

A los miembros del Departamento de Electrónica, Instrumentación y Control por su entereza y responsabilidad en el cumplimiento de sus labores.

A la facultad de Ingeniería Electrónica y Telecomunicaciones.

A la Universidad del Cauca por su contribución administrativa y económica

Resumen

El trabajo presentado a continuación “Base para sistema de entrenamiento quirúrgico: robot LapBot” hace parte de un proyecto que busca construir un sistema de entrenamiento quirúrgico para cirugía laparoscópica. Dicho proyecto consta de la construcción de dos robots, uno de ellos es un robot para cirugía laparoscópica abdominal y es operado mediante un joystick y el otro es un robot porta endoscopio encargado la visión al interior del paciente, igualmente manejado por un joystick.

En primer lugar se instaló el software, en el cual se incluía una plataforma de desarrollo y unas librerías gráficas que se requerían para el manejo de este tipo de software. Este software era necesario para la manipulación de los robots desde el computador y además para que se pudiera mostrar en pantalla los movimientos del mismo.

Para la comunicación entre el robot y el computador se diseñó una tarjeta de adquisición de datos, su funcionamiento está basado en un canal bidireccional de transmisión de información, soportada en el protocolo USB. Las posiciones articulares del robot son enviadas desde el robot hacia el computador, a su vez el computador envía la información donde está contenida la posición que se desea alcanzar con el robot.

Se desarrolló un software el cual estaría encargado de comunicar el robot con el computador, para ello se realiza un código que se embebe en el microcontrolador, el cual se encarga de leer los sensores de posición además de activar los motores.

Posteriormente se realizó el proceso de elección de los sensores, motores y demás elementos encargados de otorgar la potencia necesaria para que el sistema funcionara. Los sensores son potenciómetros lineales que permiten leer las posiciones del robot. Los motores seleccionados son motor reductores con un torque adecuado para mover las articulaciones del robot. Para encontrar los pares respectivos a cada motor se modeló en Solid Edge el LapBot, y a partir de esto se obtuvieron sus parámetros dinámicos.

Luego se procedió a hacer el diseño y corte de las piezas que conformarían el LapBot. Las piezas se fabricarían en madera, elaboradas con una cortadora láser. Posteriormente se pintaron y adecuaron para ensamblarse en cada una de las articulaciones que lo conformarían. Se implementaron una serie de elementos que otorgarían movimiento a las articulaciones pasivas y otros que mejorarían el funcionamiento de las articulaciones activas.

Después se unificó el sistema, obteniéndose la estructura propuesta. Se realizaron diferentes pruebas al hardware y software, tanto por separado como en conjunto para evaluar los resultados obtenidos.

Abstract

The document presented below "Base for surgical training system: LapBot robot" belongs to a project that search to build a surgical training system for laparoscopic surgery. This project consists of construction of both robots, one of them is a robot for abdominal laparoscopic surgery and is operated by a joystick and the other is an endoscope holder robot in charge of the internal vision of the patient, operated by joystick too.

First it was installed a software that includes the development platform and some graphics libraries that were required for the handling of this kind of software. This software was necessary for the manipulation of the robots from the computer and to show their movements in screen.

For the communication between the robots and the computer it was designed a data acquisition card, its operation is based on the transmission bidirectional channel, it is supported by usb protocol. The articular positions of the robot are send to the computer and this one sends information of the position that is desired to reach with the robot.

It was developed a software that manages the communication of the robot with the computer. For this, it was implemented a code that is embedded in a microcontroller and it reads the position sensors and activates the motors.

Later it was completed the process of selection of sensors, motors and other elements required for the system operation. The sensors are lineal potentiometers that can read the motor's position. The motors have the right torque to move the articulations. The torques were found by modeling the LapBot robot in SolidEdge where there were found the dynamic parameters.

Then it was made the cut and design of the pieces that shaped the LapBot. These pieces were made in wood by a laser cutter. Later there were painted and adjusted to be assembled in each articulation. Were also implemented different kinds of elements to improve the working of the system.

Then the system was unified and it was obtained the proposed structure. There were made different tests, software and hardware, to evaluate the obtained results.

TABLA DE CONTENIDO

Pag.

Contenido

1. Introducción	13
2. Robótica médica	15
2.1 Conceptos generales	15
2.2 Robótica quirúrgica.....	16
2.3 Laparoscopia	22
2.4 Entrenadores quirúrgicos	23
2.4.1 Simuladores para entrenamiento en cirugía laparoscópica	24
2.4.2 Simuladores para entrenamiento en cirugía robótica	26
2.5 LapBot.....	29
2.6 Hibou	31
3. Hardware involucrado en la construcción del LapBot.....	33
3.1 Estructura funcional del LapBot.....	33
3.2 Modelado matemático de LapBot	36
3.3.1 Diseño de las piezas del robot mediante software CAD	37
3.4 Selección de los motores.....	39
3.4.1 Calculo de pares	40
Figura 3.10 Pares obtenidos con Matlab al introducir los parámetros de SolidEdge.	43
3.4.1 Rediseño de los cuerpos del LapBot	44
3.4.2 Corte de las piezas del LapBot.....	47
3.4.4 Articulaciones del LapBot	48
3.5 Selección de los potenciómetros	58
3.6 Selección del puente H.....	62
3.7 Fuente de voltaje.....	63
3.8 Tarjeta de adquisición de datos	64
4. Software para la manipulación del Lapbot.....	66
4.1 Adaptación del software para el Lapbot	66
4.1.1 Integración de la librería MPUSBAPI.DLL al software para la manipulación.....	66
4.1.2 Vinculación de MPUSBAPI.DLL a Visual Studio 2008	66
4.1.3 Declaración de variables locales para utilizar la librería MPUSBAPI.DLL.....	67
4.1.4 Funciones de la librería MPUSBAPI.DLL	68
4.2 Software del microcontrolador	68

5. Pruebas y resultados	69
5.1 Pruebas hardware	69
5.2 Pruebas software.....	72
5.3 Pruebas conjuntas	73
6. Conclusiones y Trabajos futuros	78
6.1 Conclusiones	78
6.2 Trabajos futuros	78
REFERENCIAS	80
ANEXOS	85

Lista de Tablas

Pag.

Tabla 3.1 Masas de cada articulación obtenidas en SolidEdge.....	41
Tabla 3.2 Distancia a los centros de masa de cada articulación en los ejes X, Y y Z.....	41
Tabla 3.3 Primer momento de inercia de cada articulación en los ejes X, Y y Z.....	42
Tabla 3.4 Segundo momento de inercia en Kg.cm ²	42
Tabla 3.5 Medición potenciómetro deslizable (distancia – resistencia).....	59
Tabla 3.6. Medición potenciómetro deslizable con voltaje (distancia – voltaje).....	60
Tabla 3.7. Medición potenciómetro con voltaje (ángulo– voltaje).....	61

Lista de Figuras

	Pag.
Figura 2.1. Robot PUMA 560.....	18
Figura 2.2. Robot AESOP.....	18
Figura 2.3. Consola (a) y brazos quirúrgicos (b) del robot ZEUS.....	19
Figura 2.4. Da Vinci SurgicalSystem.....	20
Figura 2.5. Da Vinci Si HD SurgicalSystem.....	21
Figura 2.6. Robot Amadeus.....	22
Figura 2.7. Configuración de elementos básicos para laparoscopia (a) y (b).....	23
Figura 2.8. Entrenador para cirugía laparoscópica proMIS.....	25
Figura 2.9. Simulador con realimentación de tacto [36].....	26
Figura 2.10. Prototipo del simulador operando (a) y diferentes volúmenes de la rodilla segmentados en 3D (b) [37].....	26
Figura 2.11. Segundo prototipo de simulador de laparoscopia [33].....	27
Figura 2.12. Consola (a) y escenarios del dv-Trainer de Mimic (b).....	28
Figura 2.13. Consola (a) y escenarios del RoSS (b).....	29
Figura 2.14. Escenario de entrenamiento del SEP Robot [42].....	30
Figura 2.15. Ambiente tridimensional donde se muestra al LapBot.....	32
Figura 2.16. Hibou diseñado en CAD SolidEdge®.....	33
Figura 2.17. Casco y ambiente virtual que conforman el dispositivo auxiliar de mando del robot Hibou [45].....	33
Figura 3.1. Estructura cinemática inicial del robot LapBot [3].....	34
Figura 3.2 Tabla de los parámetros geométricos del LapBot [3].....	35
Figura 3.3. Estructura cinemática del robot LapBot rediseñado.....	35
Figura 3.4 Una de las piezas de la articulación dos.....	37

Figura 3.5 Una de las piezas de la articulación tres.....	38
Figura 3.6. Actuador articulación rotacional diseñado en Solid Edge.....	38
Figura 3.7. Estructura segunda y tercera articulación en Solid Edge.....	39
Figura 3.8. Estructura completa del LapBot en Solid Edge.....	39
Figura 3.9. Propiedades físicas de la primera articulación obtenidas con SolidEdge.....	40
Figura 3.10. Pares obtenidos con Matlab al introducir los parámetros de solidEdge.....	43
Figura 3.11. Error cartesiano obtenido con Matlab.....	44
Figura 3.12. Actuador lineal Linak.....	46
Figura 3.13. Actuador articulación de rotación importado por Dynamo Electronics.....	46
Figura 3.14. Actuador articulación final marca Hitech.....	47
Figura 3.15. Piezas del LapBot.....	48
Figura 3.16. Piezas LapBot, cuerpos uno (a), (b) y dos (c)	48
Figura 3.17. Distintas vistas de la base metálica del primer actuador.....	49
Figura 3.18. Diferentes vistas de la caja contenedora del eje del primer actuador.....	49
Figura 3.19 .Diferentes vistas del buje de extensión (a) y (b).....	50
Figura 3.20. Base con caja protectora de la segunda articulación.....	51
Figura 3.21. Base de la articulación dos conectada con la articulación uno.....	51
Figura 3.22. Motorreductor de la segunda articulación dentro de la caja.....	52
Figura 3.23. Motorreductor protegido completamente por la caja.....	52
Figura 3.24. Segunda articulación con los motores.....	53
Figura 3.25. Adaptadores hechos a la medida a y b.....	53
Figura 3.26. Segunda y tercera articulación.....	54
Figura 3.27. Tubo fijo a la cuarta articulación y que permite el inicio de la quinta articulación.....	55
Figura 3.28. Segunda parte de la quinta articulación.....	56
Figura 3.29. Sexta articulación.....	56
Figura 3.30. Séptima articulación.....	57

Figura 3.31. Distintas vistas de las articulaciones 4, 5, 6 y 7. A, b y c.....	57
Figura 3.32. Potenciómetro deslizable.....	59
Figura 3.33 .Potenciómetro Vishay.....	59
Figura 3.34. Relación voltaje – distancia.....	61
Figura 3.35. Relación voltaje - ángulo.....	62
Figura 3.36. Puente H Pololu high-power motor driver 18v15.....	63
Figura 3.37. Fuente de alimentación.....	63
Figura 3.38. Micro controlador 18F4550.....	64
Figura 3.39. Circuito impreso de la tarjeta de adquisición de datos.....	65
Figura 3.40. Esquema de la comunicación entre el computador y el robot.....	65
Figura 5.1. Potenciómetro deslizable con eles instaladas para evitar el giro del primer actuador.....	71
Figura 5.2. Buje adaptado para que funcione con el potenciómetro deslizable.....	71
Figura 5.3. Sistema completo.....	74
Figura 5.4. Posición inicial del LapBot.....	74
Figura 5.5. Movimiento hacia adelante del LapBot.....	75
Figura 5.6. Movimiento hacia atrás del LapBot.....	75
Figura 5.7. Movimiento hacia la izquierda del LapBot.....	76
Figura 5.8. Movimiento hacia la derecha del LapBot.....	76
Figura 5.9. Identificación de las dos tarjetas.....	77

1. Introducción

La robótica médica ha revolucionado la medicina contribuyendo a mejorar diversos aspectos en los campos de rehabilitación y medicina quirúrgica, tales como la ampliación de las habilidades de los médicos, la reducción del tiempo de realización de las cirugías y recuperación de los pacientes, y la mejora en la calidad de vida de estos últimos. Los primeros robots quirúrgicos fueron utilizados a partir de 1985 [1], desde ese momento se empezó a desarrollar la robótica quirúrgica como se conoce hoy en día. La técnica de operación en la que más se utilizan robots es la laparoscopia, que permite realizar operaciones sin hacer gran daño a los pacientes [2]. Las ventajas originales de la laparoscopia se multiplican cuando es realizada con ayuda de un robot [2].

Aunque la robótica quirúrgica ha avanzado notablemente y actualmente se realizan varias operaciones con robots, el robot quirúrgico más ampliamente utilizado hasta el día de hoy es el Da Vinci, que tiene muchas ventajas pero los altos costos que implican su compra y el entrenamiento para manipularlo hacen del Da Vinci un sistema de difícil acceso, especialmente para países en vía de desarrollo y para estudiantes de medicina que tendrán que enfrentarse a cirugías robóticas en el ejercicio de su profesión.

A raíz de esto, y aunque el Da Vinci es constantemente actualizado, se hace necesario estudiar y corregir ciertas funciones para lograr un acceso más universal a las ventajas de la cirugía robótica.

La robótica médica en Colombia ha tenido un desarrollo lento, sin embargo algunas universidades han logrado realizar aportes en la robótica para la rehabilitación y la cirugía. Algunas de las instituciones que han realizado estos aportes son la Pontificia Universidad Bolivariana de Medellín, la Universidad Nacional sede Medellín, la Universidad Javeriana sede Bogotá, la escuela de Ingeniería Julio Garavito y la Universidad del Cauca.

En la Universidad del Cauca el desarrollo de la robótica médica se ha dado a través del grupo de investigación en Automática Industrial que ha realizado investigación, diseño y desarrollo en las dos áreas de la robótica médica: de rehabilitación y quirúrgica.

Tratando de contribuir en la solución de la problemática de los entrenadores quirúrgicos, el trabajo presentado a continuación pretende establecer las bases para la construcción de un sistema de entrenamiento quirúrgico. La construcción de este prototipo parte del proyecto: "Modelado, Simulación en 3D y Control de un Robot para Cirugía Laparoscópica" [3], desarrollado en la Universidad del Cauca. En ese proyecto se realizó el modelado matemático y estructural de un brazo robótico llamado LapBot, además del diseño de un controlador por par calculado (CTC), para que el robot realizara una colecistectomía. En este proyecto se inicia la construcción de este brazo para realizar dicha cirugía laparoscópica con miras a ser mejorado en el futuro.

El desarrollo de este documento se inicia en el capítulo dos donde se presentan conceptos básicos sobre robótica médica, robótica quirúrgica, robots quirúrgicos, laparoscopia, entrenadores quirúrgicos y los robots quirúrgicos desarrollados en la Universidad del Cauca, el LapBot y el Hibou. En el capítulo tres se describe toda la parte hardware del proyecto,

materiales y todo lo relacionado con la construcción física del mismo. En el capítulo cuatro se presenta lo referente al software de manipulación del robot y su desarrollo y en el capítulo cinco se tienen las conclusiones obtenidas sobre el trabajo realizado y los trabajos futuros.

2. Robótica médica

La robótica empezó facilitando y mejorando el campo de la industria, transformando la forma de trabajar y producir, sin embargo los robots industriales eran solo conocidos por quienes los operaban. Ahora la robótica ha entrado en muchos aspectos de la vida de las personas, desde juguetes, robots acompañantes, robots del hogar, hasta robots que ayudan en las cirugías. Estos últimos pertenecen a la rama de la robótica quirúrgica y junto con la robótica de rehabilitación, que incluye las prótesis robóticas, exoesqueletos, robots asistentes en rehabilitación física, entre otros, forman la rama de la robótica médica.

La robótica médica se empezó a desarrollar cuando los médicos e investigadores observaron los diferentes problemas que se presentaban en los procedimientos quirúrgicos y en el desarrollo de las terapias de rehabilitación. Ellos notaron que era necesario corregir muchos errores que podían poner en riesgo la vida del paciente o demorar el proceso de recuperación. De esta forma, aparte de implementar nuevas técnicas y promover el entrenamiento del personal médico, los investigadores buscaron en la tecnología soluciones óptimas lo que llevó al desarrollo de prótesis y sistemas de recuperación robóticos y robots quirúrgicos que han mejorado notablemente la realización de tratamientos y operaciones.

2.1 Conceptos generales

Como una rama de la tecnología, la robótica maneja varios conceptos fundamentales para su desarrollo, estos son [4]:

- **Articulación:** Mecanismo que une dos cuerpos sucesivos, accionado por un motor. Las articulaciones son principalmente rotoides (de giro) o prismáticas (de desplazamiento). Aunque existen articulaciones pasivas (sin motor) que reproducen cualquiera de los dos movimientos.
- **Articulación rotoide:** El movimiento de rotación se realiza alrededor de un eje común entre dos cuerpos.
- **Articulación prismática:** El movimiento de traslación se realiza a lo largo del eje común entre dos cuerpos.
- **Grado de libertad:** Define cada movimiento independiente del robot. Para situar un objeto en un espacio tridimensional son necesarios tres grados de libertad, uno por cada dimensión.

- **Espacio articular:** Es el espacio en el cual se representa la situación de todos los cuerpos del robot; corresponde al lenguaje que maneja el mecanismo en sí mismo (movimientos rotacionales o prismáticos).
- **Espacio operacional:** Es aquel donde se representa la situación del órgano terminal. Para definir esta situación se utilizan las coordenadas cartesianas en tres dimensiones. Es llamado también espacio cartesiano.
- **Modelo geométrico Directo:** Este modelo expresa la situación del órgano terminal, es decir, las coordenadas operacionales del robot. Este modelo permite conocer la posición cartesiana y la orientación del órgano terminal si se conocen previamente las posiciones articulares.
- **Modelo geométrico Inverso:** Entrega todas las posibles soluciones de las posiciones articulares para una posición y orientación deseadas del órgano terminal con el fin de alcanzar esa situación deseada.
- **Modelo dinámico:** El modelo dinámico contiene toda la información (geométrica y dinámica) del robot. Es utilizado para la simulación en computador, el dimensionamiento de los actuadores, la identificación de los parámetros inerciales y de frotamiento del robot y el control.
- **Modelo dinámico directo (MDD):** Expresa las aceleraciones articulares en función de las posiciones, velocidades y fuerzas en las articulaciones.
- **Modelo dinámico inverso (MDI):** Se define como la relación entre las fuerzas aplicadas a los actuadores y las posiciones, velocidades y aceleraciones articulares.

2.2 Robótica quirúrgica

En 1985 el robot PUMA 560 (figura 2.1) se utilizó para guiar una biopsia cerebral y en 1988 se utilizó el PROBOT que realizaba una imagen tridimensional de la próstata para que el cirujano supiera donde hacer los cortes. Sin embargo no fue sino hasta 1992 que IBM desarrolló el primer robot autorizado por la FDA, el ROBODOC que fue utilizado en cirugías para reemplazo de cadera [1] y [5]. En 1994 la FDA aprobó el robot AESOP 1000 (figura 2.2), que fue el primer robot que realizó una cirugía laparoscópica pues reemplazaba al ayudante que sostenía la cámara laparoscópica, esto hizo que se redujera el número de personas en la sala de cirugía y aún se utiliza en diversos centros médicos [5] y [6].



Figura 2.1. Robot PUMA 560 [1] y [5].



Figura 2.2. Robot AESOP [5] y [6].

Después de AESOP, el departamento de defensa de los Estados Unidos impulsó el proyecto MASH (*Mobile Advanced Surgical Hospital*) para realizar operaciones a distancia en el campo de batalla, sin embargo la cantidad de señales que se tiene en

una zona de batalla es muy alta lo que produce grandes interferencias que hacen imposible el control del robot que va a realizar la operación [1] y [6].

A partir de estos proyectos se empezó la investigación de los robots quirúrgicos “on-line” que son los que siguen los movimientos del cirujano en tiempo real evitando el temblor y los errores. Estos se dividen en dos tipos: los manipuladores endoscópicos como el AESOP y los sistemas amo-esclavo [7] y [8]. Estos últimos se caracterizan por tener una consola quirúrgica con dos o tres brazos robóticos que sirven para manejar los instrumentos. Los robots característicos de este tipo son el ZEUS (figura 2.3 a y b) y el Da Vinci (figura 2.4). Este último es el más moderno y el más utilizado actualmente por más de 3500 cirujanos [9].

El robot ZEUS fue creado en 1998 por la compañía ComputerMotion de California con el propósito de realizar cirugías a distancia con instrumentos laparoscópicos, su plataforma se monta sobre la mesa de operaciones y es modular, donde uno de sus módulos es el robot AESOP que opera la cámara. Tiene 5 grados de libertad y su bajo peso lo hace portable [10].

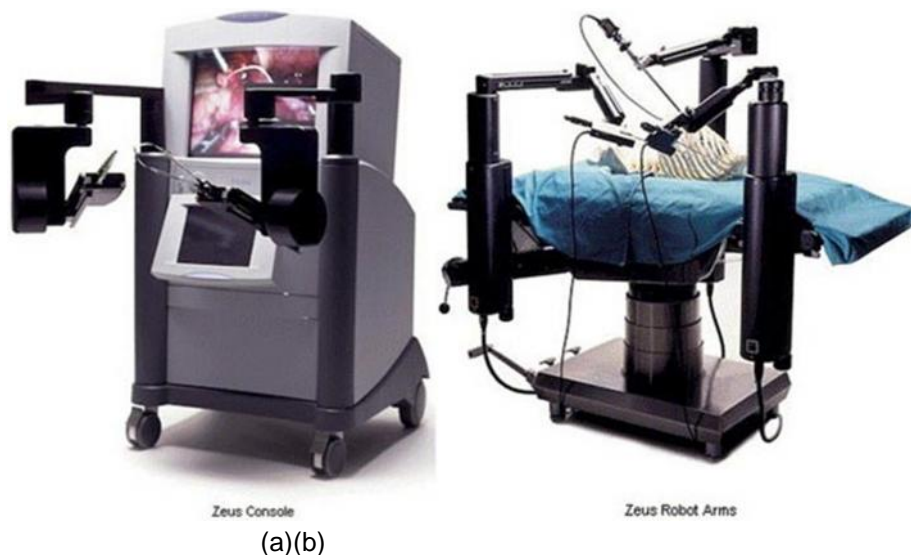


Figura 2.3. Consola (a) y brazos quirúrgicos (b) del robot ZEUS [10].

Pocos meses después de la salida del ZEUS se creó el sistema robótico con mayor éxito comercial hasta el momento, el Da Vinci. Fue desarrollado por Intuitive Surgical también de California, está compuesto por tres o cuatro brazos, uno que sostiene la cámara y los otros que operan, se monta sobre el suelo, no es modular y pesa un poco más de media tonelada [10].

El Da Vinci es actualmente el sistema robótico para cirugías laparoscópicas más utilizado a nivel mundial y el único en el mercado, es actualizado constantemente y ofrece enormes ventajas tanto al médico como al paciente. El paciente puede tener

una operación más sencilla, menos dolorosa y con una recuperación más rápida, lo que permite que esté menos tiempo en el hospital. Para el cirujano también existen varias ventajas pues amplifica las habilidades del cirujano permitiendo que sus movimientos sean más exactos filtrando el temblor que se da después de varias horas de operación. Una de las características del Da Vinci es que no es programado, solo refleja los movimientos del cirujano por lo que requiere de un equipo médico de soporte durante las cirugías y posee un sistema de apagado seguro en caso de que se interrumpa el flujo eléctrico, lo que permitiría que el cirujano termine la operación manualmente [11].



Figura 2.4. Da Vinci SurgicalSystem [9].

La actualización del año 2008 del sistema quirúrgico da Vinci, el Da Vinci S HD SurgicalSystem, le da una capacidad de visión de la imagen obtenida en alta definición con la plataforma robótica, además maneja un zoom digital que reduce las interferencias entre el endoscopio y los instrumentos. Es el primer sistema en tener visión 3D en alta definición lo que hace que los ojos y las manos del cirujano se extiendan virtualmente hasta el paciente [12]. Todo lo anterior se suma a las características generales del Da Vinci que han hecho de este el mejor robot quirúrgico construido hasta ahora, algunas de ellas son: un carro que se sitúa junto al paciente y que a su vez está conectado a una consola central, de tres a cuatro brazos manipuladores robóticos y obtención de imágenes a través de la lente de la cámara laparoscópica, para posteriormente ser integradas por el ordenador y así crear una imagen tridimensional (3D) [9],

En abril del 2009, IntuitiveSurgical lanza la última y más moderna versión del Da Vinci, el Da Vinci Si HD SurgicalSystem que mejora la anterior actualización e introduce nuevas características como la posibilidad de instalar una segunda consola que

permite realizar operaciones colaborativas y facilitar la enseñanza y el entrenamiento de cirujanos. La interfaz de usuario se manipula desde una pantalla táctil que le permite al cirujano tener un control preciso de las características de video, audio y de los ajustes del sistema [13] y [14].



Figura 2.5. Da Vinci Si HD SurgicalSystem [9].

El robot Da Vinci ha mejorado notablemente los procedimientos de laparoscopia al optimizar la visibilidad, anular el temblor, elevar la precisión quirúrgica y hacer más cómodo el procedimiento para el cirujano al proporcionarle un carro más ergonómico que le permite operar en una mejor posición[15]. Sin embargo su principal problema es el elevado costo de más de un millón de dólares, lo que lo hace muy difícil de adquirir por la mayoría de los hospitales del mundo [15] y [16]. Además del precio se tiene en cuenta que los costos de operación con el robot son muy altos y requiere personal y espacio especial para su correcta manipulación [16]. Además, a pesar de haber mejorado la visión 3D, no existe retroalimentación táctil, lo que da la sensación de insensibilidad a los cirujanos y puede afectar el desempeño de aquellos con poca experiencia [17]. A esto se le agrega que al ser una única empresa la que tiene al único robot cirujano en el mercado se crea un monopolio que no permitirá disminuir el precio del Da Vinci ni comparar sus características con otros robots cirujanos.

Esto dificulta el aprendizaje de los cirujanos que están interesados en las cirugías robóticas, pues para aprender a utilizar el Da Vinci se debe manejar directamente al Da Vinci, se conocen pocos simuladores o entrenadores en cirugía robótica que permitan a los cirujanos aprender a operar con un robot antes de realizar una intervención real y aunque el Da Vinci Si HD SurgicalSystem dispone de dos consolas que pueden adaptarse para que un cirujanos entrene a otro, el hospital donde se realice el

entrenamiento debe adquirir el robot y costear los gastos que se generen en el entrenamiento. La mejor forma conocida de entrenamiento es la enseñanza directa. El curso más conocido es el dictado en el WorldLaparoscopy Hospital, lo han cursado más de 400 cirujanos, sigue el método de enseñanza médica donde un cirujano muestra las diferentes técnicas y los alumnos observan y los cirujanos tiene la oportunidad de manipular el da Vinci durante el curso pero nada asegura que vayan a continuar practicando pues es posible que no tengan acceso al Da Vinci, además el curso tiene un costo de 5.000 dólares para cirujanos que no son de la India y de aproximadamente 2.800 dólares para los que viven en el país [18].

En Canadá, la empresa Titan Medical Inc. ha desarrollado un robot quirúrgico de cuatro brazos llamado Amadeus (figura 2.6). Este robot es el primer sistema de cirugía al que se le pueden colocar varios instrumentos multi-articulados y una cámara 3D articular integrada con iluminación, todo a través de una incisión de 25 mm. Además tiene comunicación y sistema de visión mejorados y de respuesta de fuerza. Su sistema quirúrgico se llama SPORT (*single portorificerobotictchnology*), está en desarrollo continuo y tiene una plataforma de visión única controlada por el cirujano. El diseño contempla un dispositivo plegable que cuando esta doblado puede ser introducido al paciente, una vez adentro podrá ser controlado por el cirujano a través de una configuración determinada [19].



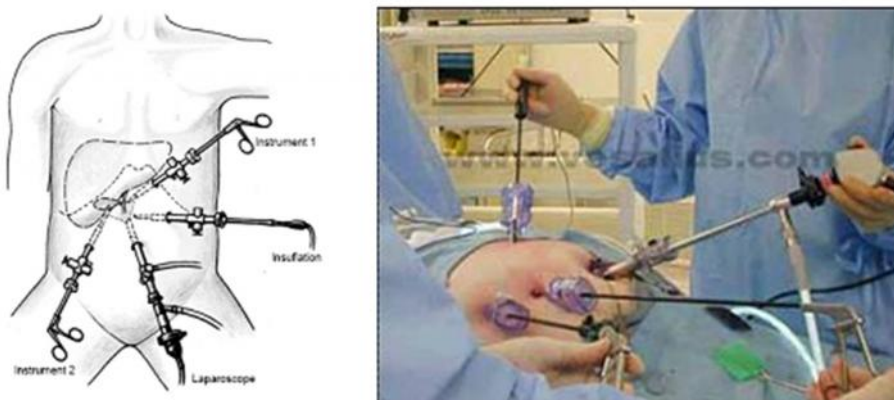
Figura 2.6. Robot Amadeus [19].

En Colombia la cirugía robótica se realiza, desde 2009, en la Clínica Marly de Bogotá que es el único lugar que tiene un Da Vinci en el país. En el proyecto están involucrados médicos urólogos, ginecólogos, cirujanos generales, cirujanos cardiovasculares y anestesiólogos que ofrecen servicios en cirugía de tórax, urología, cardiovascular, general y ginecología [20]. En una entrevista realizada en febrero de 2013 por el periódico Vanguardia Liberal de Santander el director médico de urología de Cirurobótica de la Clínica Marly, Camilo Giedellman, afirmó que hasta ahora se han realizado con el Da Vinci 150 operaciones y el jefe del servicio de Urología de la

Clínica Marly y gerente general de la Institución, Luis Eduardo Cavellier, indicó que el costo de las cirugías varía entre los 25 y los 35 millones de pesos. En febrero de 2013 el médico urólogo pediatra Pablo Gómez Herrera realizó exitosamente con este robot por primera vez en Colombia y en Latinoamérica una cirugía a un niño de dos años que tenía una anomalía en la uretra [21].

2.3 Laparoscopia

La laparoscópica es un procedimiento en el cual, a través de pequeñas incisiones, el experto introduce un endoscopio en el abdomen obteniendo una imagen de los órganos en un monitor [22]. A través de estas incisiones el médico procede introduciendo por los trocares diferentes elementos tales como pinzas, tijeras, cauterizadores, entre otros, junto con el endoscopio [23]. La principal ventaja de la cirugía laparoscópica es que permite ejecutar procedimientos sin realizar grandes cortes al paciente lo que genera un impacto psicológico positivo al tener menos traumatismos estéticos, reduce el tiempo de recuperación, el dolor y el riesgo de infecciones [24].



(a) (b)

Figura 2.7. Configuración de elementos básicos para laparoscopia (a) y (b).

La cirugía laparoscópica está ampliamente extendida por las múltiples ventajas que presenta y los procedimientos más comunes realizados con esta técnica son:

- **Apendicectomía:** Es el procedimiento más común y consiste en la extirpación de un apéndice inflamado o infectado (apendicitis). Para llevarla a cabo se hace una incisión pequeña en el cuadrante inferior derecho del abdomen [25].
- **Colecistectomía:** Cuando la vesícula biliar está inflamada u obstruida por cálculos biliares se realiza esta intervención para quitarla [26].

- **Colectomía asistida:** Extracción o eliminación de una parte enferma del intestino grueso (colon)[23].
- **Gastrectomía:** Al presentarse problemas gástricos crónicos como úlceras o cáncer los médicos se ven en la necesidad de extirpar parcial o totalmente el estómago [27].
- **Hepatectomía:** Esta intervención es un medio para determinar, conocer y retirar las lesiones y tumores en el hígado [28].
- **Histerectomía:** El cirujano realiza una incisión por la cual corta las trompas de Falopio y separa el útero (matriz) en su unión con el cuello para poder extirparlo [29].
- **Ligadura de trompas:** Se cierran las trompas de Falopio de una mujer como una forma permanente de evitar el embarazo [30].
- **Prostatectomía:** Intervención para la curación del cáncer, la preservación de la continencia o la función sexual. Se extrae la totalidad o parte de la glándula prostática (próstata)[31].
- **Vasectomía:** Procedimiento que consiste en cortar y ligar los conductos deferentes, impidiendo el recorrido del semen y así lograr la esterilización masculina[32].

2.4 Entrenadores quirúrgicos

Se sabe que una de las principales desventajas de la cirugía laparoscópica es la falta de entrenamiento que tiene el médico que se va a enfrentar por primera vez a esta operación. Para solucionar este problema existe la propuesta de realizar entrenadores quirúrgicos que permitan que el médico se familiarice con los procedimientos que va a realizar y de esta forma evitar utilizar cirugías reales como método de enseñanza pues esto podría implicar un peligro para el paciente.

Los simuladores quirúrgicos deben cumplir ciertas características en cuanto al realismo de los órganos pues estos deben responder a las interacciones con el cirujano, por ejemplo deformándose, y reaccionando mediante modificaciones estructurales a acciones quirúrgicas habituales como cortes, cauterización o sutura.

Cada simulación de una cirugía con una interfaz tiene diversos retos como la geometría de los órganos, la apariencia y la dinámica de cada uno (sangrado, elasticidad, movimiento de nervios).

2.4.1 Simuladores para entrenamiento en cirugía laparoscópica

La búsqueda de realismo en los entrenadores quirúrgicos hace que cada vez se requieran más ordenadores y programas grandes y complejos. El desarrollo de simuladores para cirugía laparoscópica ha sido grande y se encuentran variedad de soluciones que aunque sirven para el médico no están enfocadas en la cirugía robótica. Estos permiten al médico aprender más rápido, escoger el tipo de cirugía a realizar que puede ser sencilla o de rara ocurrencia, y al realizarse el aprendizaje en un simulador se disminuirán los procedimientos en personas y animales [33].

Los entrenadores de cirugía laparoscópica son especialmente útiles cuando el procedimiento tiene más riesgos por lo que se requiere mayor dedicación siendo la curva de aprendizaje más lenta. Generalmente el entrenamiento quirúrgico está orientado a adquirir habilidad para la práctica y evitar complicaciones o efectos adversos [34].

Los simuladores para entrenamiento en cirugía laparoscópica se caracterizan por ser o únicamente virtuales o tener una parte real y otra virtual. La parte real en general se trata de un maniquí de simulación de reanimación cardiopulmonar al cual se le introducen los instrumentos. Con base en esto se creó el simulador proMIS (figura 2.8) de la empresa Háptica para entrenar cirujanos, tiene cinco módulos para laparoscopia, manejo de instrumentos, disección, diatermia y suturado [35].



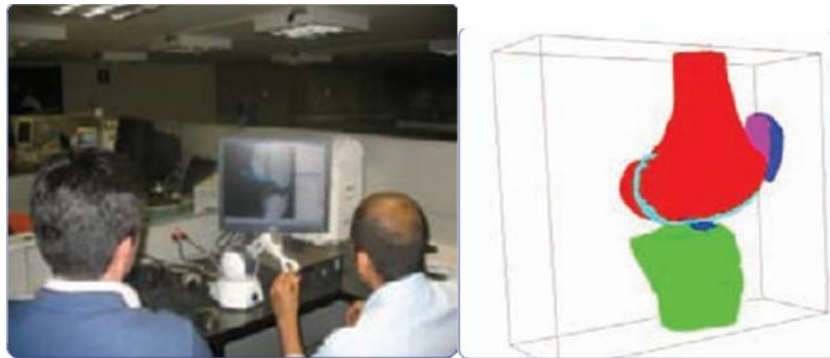
Figura 2.8. Entrenador para cirugía laparoscópica proMIS [35].

Otro entrenador importante es el diseñado por Delingette y Ayache que simula una cirugía hepática y tiene la característica de darle la sensación al cirujano de que está tocando el órgano por medio de una interfaz háptica [36], esto le da una mejor idea de cómo son los órganos reales.



Figura 2.9. Simulador con realimentación de tacto [36].

En Colombia hasta el 2008 no se tenían referentes sobre diseño o implementación de simuladores para entrenamiento quirúrgico hasta que ingenieros estudiantes de maestría y médicos de la Universidad de los Andes, la Universidad del Bosque y la fundación Santa Fe desarrollaron un sistema de entrenamiento médico para artroscopia de rodilla que consiste en una rodilla en 3D simulada por computador y de un dispositivo de retorno de fuerza que produce la sensación de un procedimiento artroscópico real, el modelo de la rodilla se reconstruye a partir de imágenes tomadas de una resonancia magnética hecha al paciente [37].



(a) (b)

Figura 2.10. Prototipo del simulador operando (a) y diferentes volúmenes de la rodilla segmentados en 3D (b) [37].

Por otro lado en el 2010 estudiantes el programa de Ingeniería Biomédica de la Escuela de Ingeniería de Antioquia y la Universidad CES, desarrollaron el prototipo de un simulador de laparoscopia que después fue corregido al mejorar la iluminación, la calidad del video y la cavidad por donde se manipulaban los instrumentos quirúrgicos, pasando de la caja de madera

original a un maniquí de simulación de reanimación cardiopulmonar (figura 2.11). Este prototipo fue utilizado además para evaluar a estudiantes de postgrado de Clínica en Cirugía Laparoscópica Avanzada de la Universidad CES y quienes pasaron la primera prueba de habilidad de ubicación espacial y destreza con los instrumentos laparoscópicos fueron seleccionados para un futuro proceso de formación. De esta manera se muestra una nueva aplicación para los simuladores: la evaluación [33].

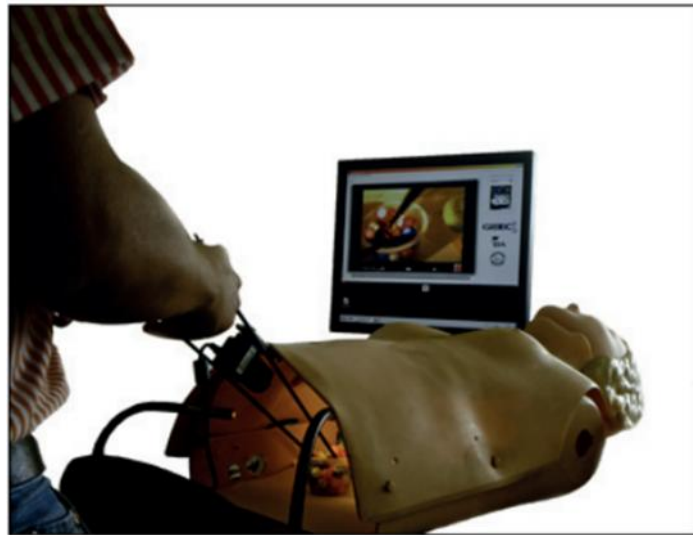


Figura 2.11. Segundo prototipo de simulador de laparoscopia [33].

2.4.2 Simuladores para entrenamiento en cirugía robótica

La mayoría de los entrenadores quirúrgicos sirven para enseñar al médico como realizar un procedimiento de forma convencional, esto es solo con cirugía mínimamente invasiva, lo que deja a un lado el entrenamiento para cirujanos que quieran aprender a manipular un robot quirúrgico.

El entrenamiento en cirugía robótica se conforma por el entrenamiento con el sistema y el entrenamiento de procedimientos. Con el incremento de la calidad de simuladores de realidad virtual para entrenamiento en cirugía robótica se espera que este tipo de entrenadores se vuelvan muy importantes para los futuros cirujanos robóticos. Los programas de entrenamiento en cirugía robótica son basados en competencias y se usan para compartir información o aprender nuevas habilidades. Algunos son solo didácticos y otros son prácticos pero muchos combinan las dos cosas [38].

Los robots quirúrgicos pueden ser utilizados para practicar todo tipo de operaciones ya sea con sistemas de entrenamiento de realidad virtual o modelos de los órganos que generan una respuesta al ser afectados por los instrumentos quirúrgicos. También se ha utilizado la cirugía por telepresencia para entrenar cirujanos, en este caso el profesor puede estar a

distancia enseñando a sus estudiantes como es una operación y estos podrían intervenir en el simulador [39].

Existen varios simuladores que permiten a los cirujanos practicar cuantas veces deseen las operaciones que se realizan con un robot. La empresa Mimic Technologies, Inc. [36] ha desarrollado un sistema llamado dv-Trainer que es un simulador que permite desarrollar las habilidades necesarias para manipular el robot Da Vinci ya que esta empresa está asociada a IntuitiveSurgical desde 2003 y su sistema de simulación no es solo virtual sino también físico pues posee una consola similar a la del Da Vinci en donde el cirujano se va a sentir cómodo y se familiarizará con el robot cirujano [40].

En agosto de 2013 Mimic Technologies, Inc. lanzó la última versión del entrenador llamada MSim 2.1 (figura 2.12 a y b) la cual tiene escenarios altamente reales con modelos 3D virtuales muy fieles a los originales pues están basados en los instrumentos del Da Vinci, posee flexibilidad para incorporar nuevos instrumentos y técnicas a medida que vayan siendo desarrolladas, también guarda el progreso y el historial de aprendizaje y crea currículos avanzados que pueden ser compartidos con otros cirujanos o instituciones para ampliar el aprendizaje [40].



(a) (b)

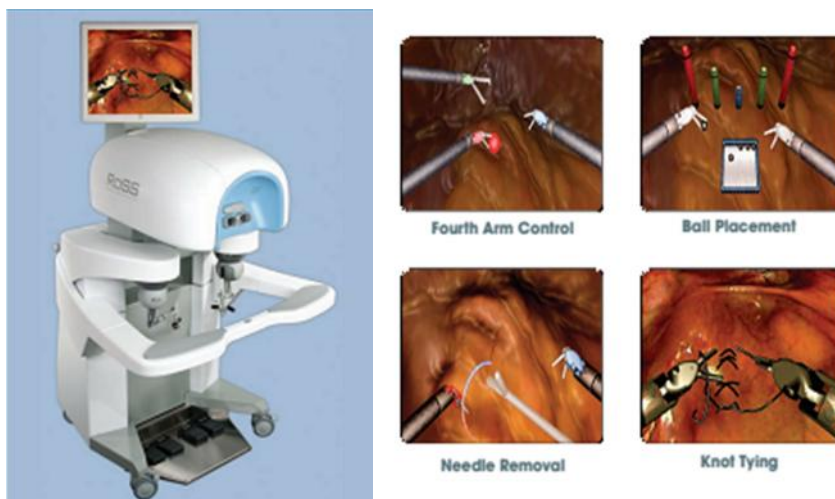
Figura 2.12. Consola (a) y escenarios del dv-Trainer de Mimic (b) [40].

De la asociación de IntuitiveSurgical con Mimic Technologies, Inc. se creó el Da Vinci Si Skills Simulator que combina el software de Mimic con la consola original del Da Vinci y es ofrecido en la página de IntuitiveSurgical, sin embargo no tiene todos los ejercicios y las tecnologías del MSim 2.1.

Otro sistema de simulación de cirugía asistida por robot es el RoSS (*RoboticSurgical Simulator*) que es un sistema de entrenamiento que utiliza

realidad virtual para introducir al usuario a los fundamentos de la cirugía asistida por robot. Es un simulador independiente y portable que enseña a los cirujanos novatos las técnicas motoras y cognitivas necesarias para operar el robot da Vinci. El RoSS tiene un currículo multinivel diseñado con varios niveles de dificultad para que el usuario vaya aprendiendo progresivamente mientras mejora sus habilidades y desarrolla mayor destreza. Tiene la capacidad de simular procedimientos quirúrgicos de larga duración en 3D [41].

El RoSS está equipado con un software avanzado, diseñado para que los cirujanos novatos conozcan el mundo virtual de la robótica con lo cual serán capaces de adaptarse a la sensación y las funciones de la consola principal del Da Vinci. El hardware del RoSS (figura 2.13 a y b) está conformado por una pantalla táctil donde se muestran los órganos simulados y todos los movimientos que realiza el cirujano y desde la cual se pueden manipular las bases de datos. Además cuenta con una consola similar a la del Da Vinci que contiene un monitor donde le muestra al cirujano en alta resolución y en 3D sus acciones, sus controles son de seis grados de libertad, tiene pedales y pinzas especiales para las versiones Da Vinci S y Da Vinci Si y tiene ruedas durables para aumentar la portabilidad [41].



(a) (b)

Figura 2.13. Consola (a) y escenarios del RoSS (b) [41].

El SEP Robot es el tercer simulador de cirugía robótica disponible en el mercado, su software de realidad virtual tiene módulos especializados de enseñanza en cirugías mínimamente invasivas como la colecistectomía laparoscópica, la laparoscopia en embarazos ectópicos, nefrectomía y ovariectomía laparoscópica. Cada tipo de cirugía tiene varios ejercicios que ayudan al cirujano a familiarizarse con el procedimiento robótico. Este

simulador no tiene consola, solo un monitor, un procesador y una placa para descansar el brazo [42] y [43].

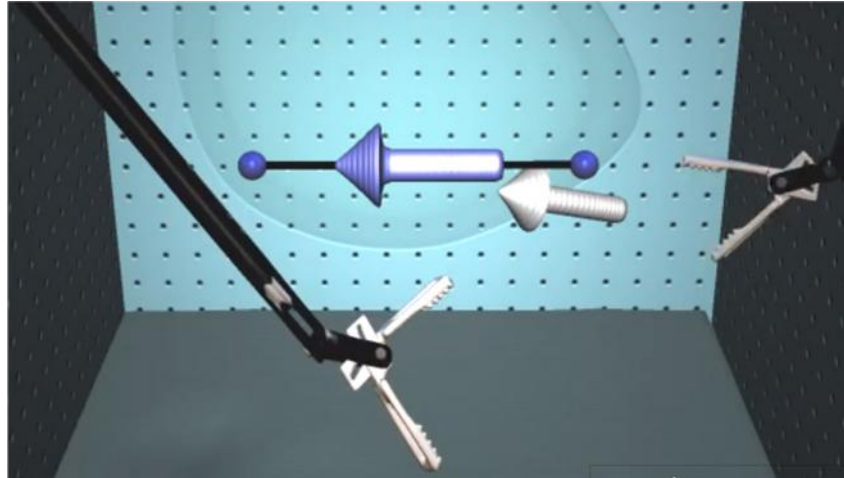


Figura 2.14. Escenario de entrenamiento del SEP Robot [42].

Como se muestra con los simuladores mencionados, al existir solo un robot aprobado para cirugías, las empresas productoras de los simuladores han desarrollado sus productos para que el cirujano aprenda a manipular este único robot, lo que sigue beneficiando directamente el monopolio de Intuitive Surgical con el da Vinci. Asimismo, aunque las empresas desarrolladoras afirmen que sus simuladores son los mejores y que tienen muchos testimonios de los clientes, aún no se han realizado estudios y validaciones multi institucionales como tampoco comparaciones de cuál de los simuladores es el mejor [43].

Los simuladores de cirugía asistida por robot no tiene precios tan exorbitantes como el Da Vinci, siendo el más costoso el RoSS cuyo precio oscila entre los \$95.000 y los \$125.000 dólares, seguido por el dV-Trainer que varía entre \$85.000 y \$100.000 dólares y el menos costoso que es el SEP Robot con precios entre los \$40.000 y los \$45.000 dólares [42].

2.5 LapBot

En el año 2009 el ingeniero Sergio Salinas desarrolló el proyecto de maestría “Modelado, Simulación en 3Dy Control de un Robot para Cirugía Laparoscópica” [3] en el cual fue diseñado el robot LapBot (Robot para Laparoscopia).

El LapBot fue diseñado basado en el análisis de las estructuras de robots como el Da Vinci, Zeus, Endobot, Black Falcon, Endovia, entre otros. En este proyecto se diseñó y sintonizó el control por par calculado (CTC) para verificar la dinámica del

LapBot y demostrar su utilidad en el seguimiento de trayectorias de prueba y del proceso quirúrgico de una colecistectomía (cirugía mini-invasiva para extraer una vesícula biliar enferma).

Para el sistema robótico LapBot se desarrolló un ambiente tridimensional en el programa Ogre3D que permite la comunicación cliente-servidor con Matlab, el cual fue el programa utilizado para realizar todas las pruebas con el controlador.

La estructura que se escogió para el diseño del LapBot con base en el estudio de los otros robots tiene nueve grados de libertad, con seis articulaciones activas que permiten posicionar y orientar el instrumento quirúrgico en un espacio tridimensional; y tres pasivas, que permiten mantener el punto fijo de movimiento sobre el punto de incisión donde se encuentra el trocar. Con estos 3 grados de libertad se garantiza que el robot es intrínsecamente seguro y que no dañará al paciente por un esfuerzo agresivo en la incisión.

Las tres primeras articulaciones permiten posicionar al LapBot en un espacio tridimensional, la cuarta articulación se fija en un valor determinado para inclinar al robot y ubicarlo por encima del abdomen del paciente, las articulaciones quinta y sexta son pasivas y mantienen el movimiento del cuerpo a través del trocar, y las últimas tres articulaciones forman una estructura en forma de muñeca que permite la orientación de la herramienta al interior del abdomen.

El LapBot fue diseñado originalmente para que esté sobre una base en el suelo, al lado de la camilla. Esta base es de 80 centímetros de alto y el brazo del LapBot, que está formado por varios cuerpos es de 95 centímetros, de los cuales los últimos 30 centímetros están casi totalmente retraídos lo que hace que el largo desde la base hasta el abdomen del paciente sea de 65 centímetros.

La pieza de 30 centímetros pasará a través del trocar hasta el interior del abdomen del paciente, el trocar puede medir entre 50 milímetros y un centímetro.

Durante el desarrollo de este proyecto también se realizó el estudio de las trayectorias quirúrgicas involucradas en una colecistectomía, esto con el fin de realizar la simulación de esta intervención con el LapBot y de esta forma probar su funcionamiento.

Se demostró que el LapBot respeta el punto de incisión y que realiza la simulación de trayectorias con un error mínimo.

El ambiente tridimensional donde se muestra la simulación del LapBot fue construido en Ogre3D y Visual C++. Este ambiente incluye dos brazos del robot LapBot, una camilla, una caja que simula el abdomen insuflado del paciente, una habitación y se diseñó una interfaz gráfica de usuario (GUI). La interfaz que incluye el proyecto permite seleccionar entre cuatro trayectorias parciales y la operación

final, estas son: Disección de Calot, Conducto cístico, Arteria cística, Lecho hepático, y Colectomía que contiene a las demás en secuencia.

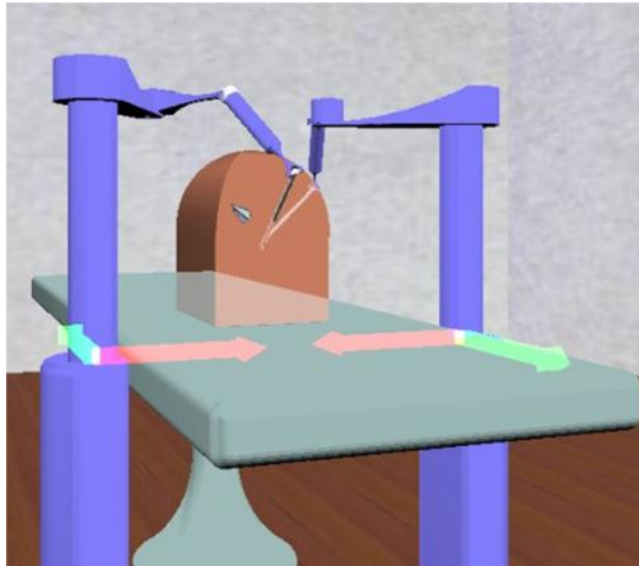


Figura 2.15. Ambiente tridimensional donde se muestra al LapBot.

2.6 Hibou

En el 2010 se realizó la tesis de pregrado llamada “Diseño y Simulación en 3D de un Robot porta endoscopio para cirugía laparoscópica” [44] por los entonces estudiantes Cristian Méndez y Víctor Torres. En este proyecto se desarrolló el Hibou (Robot porta endoscopio) con base en el estudio de las estructuras de varios robots como el AESOP, EndoAssist, LER (Light Endoscopeholder Robot), Estados Unidos, Reino Unido y Francia, y el Kirubot y LapBot de Colombia.

El modelo matemático geométrico y dinámico del Hibou fue realizado para comprobar que puede utilizarse en procedimientos laparoscópicos manteniendo un punto fijo en el espacio (trocar), que sirve como entrada a la cavidad abdominal y como punto fijo de rotación del instrumento que se introduce en el paciente. El Hibou cuenta con siete articulaciones, cinco activas y dos pasivas.

El control por par calculado (CTC) fue diseñado para trayectorias ordenadas en tiempo real desde un dispositivo exterior al PC, como el Joystick.

La interfaz realizada en este proyecto fue hecha con Matlab y el software CAD SolidEdge, en esta se observa al Hibou realizando movimientos con su órgano terminal

introducido en un paciente, a través del punto de incisión en el abdomen (trocar), y manteniendo este punto como punto fijo de rotación.

En el año 2011 se realizaron también en la Universidad del Cauca dos proyectos que en conjunto buscaban desarrollar un sistema virtual para el posicionamiento del robot Hibou. El sistema virtual fue hecho en Ogre3D y la posición del endoscopio es controlada por un casco que porta el cirujano. Dicho casco está formado por un sistema electrónico basado en un acelerómetro, el cual permite mover y orientar el órgano terminal al interior del paciente [45] y [46].



Figura 2.16. Hibou diseñado en CAD SolidEdge®.



Figura 2.17. Casco y ambiente virtual que conforman el dispositivo auxiliar de mando del robot Hibou [45].

3. Hardware involucrado en la construcción del LapBot

La construcción de la estructura física del robot LapBot, se distribuyó de la siguiente manera:

- Obtención de pares.
- Selección de elementos.
- Diseño de las piezas del LapBot.
- Corte de las piezas.
- Ensamblaje.

Los criterios de selección para cada uno de estos ítems, varían dependiendo de los requerimientos que se tenían establecidos de antemano y de los ajustes que fue necesario realizar. Se trató de buscar los elementos que más se adaptaran tanto al presupuesto como a los resultados que se deseaban obtener en el trabajo final. Se debió además, considerar el material con el cual se iba a construir el prototipo, ya que este era determinante a la hora de obtener el torque necesario en los motores para que movieran la estructura sin mayor inconveniente.

Por otro lado se debía garantizar que los motores tuvieran la potencia suficiente para funcionar según los requerimientos. Como la posición de cada una de las articulaciones debería estar determinada se usaron sensores de posición y se diseñó una tarjeta de adquisición de datos.

3.1 Estructura funcional del LapBot

Para el diseño de la estructura funcional del LapBot inicialmente se había tenido en cuenta algunas topologías estructurales existentes, definiendo una serie de requisitos mínimos que deben ser cumplidos por este robot [3], los cuales se conservan en este trabajo. Dichos requisitos mínimos, fueron la base para el desarrollo de este proyecto, en el cual se realizaron una serie de modificaciones en la estructura del LapBot porque se debían tener en cuenta algunos aspectos a la hora de la construcción del prototipo, además lo que se buscaba inicialmente era establecer una base para que en un futuro se construya un sistema de entrenamiento quirúrgico en la Universidad del Cauca. Después de este análisis se estableció que el LapBot a construir tendría:

- Siete grados de libertad en total, que constituyen una estructura que va por fuera del paciente.

- Cuatro grados de libertad activos, que permitan orientar y posicionar la herramienta en cualquier lugar dentro de un espacio tridimensional, por fuera de la incisión.
- Tres grados de libertad pasivos, para no perder la referencia manteniendo un centro de rotación remoto, que es donde estará ubicado el trocar.

Para sentar las bases de un sistema de entrenamiento quirúrgico se decidió descartar la pieza que había sido diseñada para ir dentro del paciente a través del trocar, que constaba de dos articulaciones activas y permitía orientar de forma fina la herramienta en la cavidad abdominal del paciente. Así finalmente la estructura del LapBot tendría siete grados de libertad, con cuatro articulaciones activas para orientar el instrumento en el espacio tridimensional; y tres pasivas para efectos de referencia, y que logre mantener un punto fijo, en este caso el trocar. Con estos últimos grados de libertad pasivos se garantiza que el robot es intrínsecamente seguro y que no dañara al paciente por un esfuerzo agresivo en la incisión [3].

Como se observa en la figura 3.1 originalmente la estructura cinemática del LapBot[3], tenía nueve grados de libertad, de los cuales seis llevaban motor y los tres restantes se basaban en mecanismos para mantener el trocar como punto de referencia. Las distancias están simbolizadas con las letras D y R según sus parámetros geométricos mostrados en la figura 3.2[3]. Cada una de las articulaciones están numeradas; la de translación está representada gráficamente con un prisma y las de rotación con cilindros.

El punto de incisión por donde cruza el trocar está representado con un anillo y se ubica después de la articulación siete.

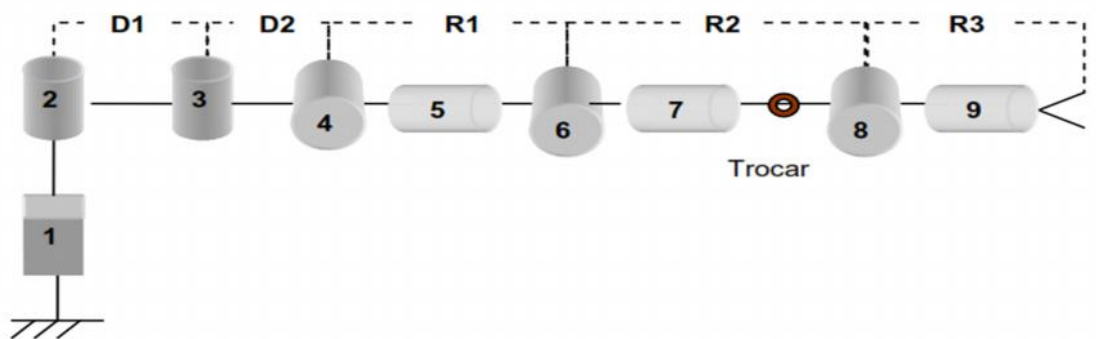


Figura 3.1. Estructura cinemática inicial del robot LapBot [3].

j	μ_j	σ_j	α_j	d_j	θ_j	r_j
1	1	1	0	0	0	r_1
2	1	0	0	0	θ_2	0
3	1	0	0	$D1$	θ_3	0
4	0	0	90°	$D2$	θ_4	0
5	0	0	90°	0	θ_5	$R1$
6	0	0	-90°	0	θ_6	0
7	1	0	90°	0	θ_7	$R2$
8	1	0	-90°	0	θ_8	0
9	1	0	90°	0	θ_9	0

Figura 3.2 Tabla de los parámetros geométricos del LapBot [3].

De la figura anterior [3] se tiene que:

j : representa el número de la articulación o cuerpo.

μ_j : indica si la articulación es activa (con motor) (1) o pasiva (sin motor) (0).

σ_j : indica si la articulación es de traslación (1) o de rotación (0).

α_j , θ_j : son ángulos que dependen de los ejes de las articulaciones o de la rotación de una articulación rotoide.

d_j , r_j : son distancias entre los ejes de las articulaciones o el desplazamiento de una articulación prismática.

Después del rediseño del LapBot, la nueva estructura cinemática se puede observar en la figura 3.3 donde las articulaciones azules uno, dos, tres y siete son activas y las moradas cuatro, cinco y seis son pasivas.

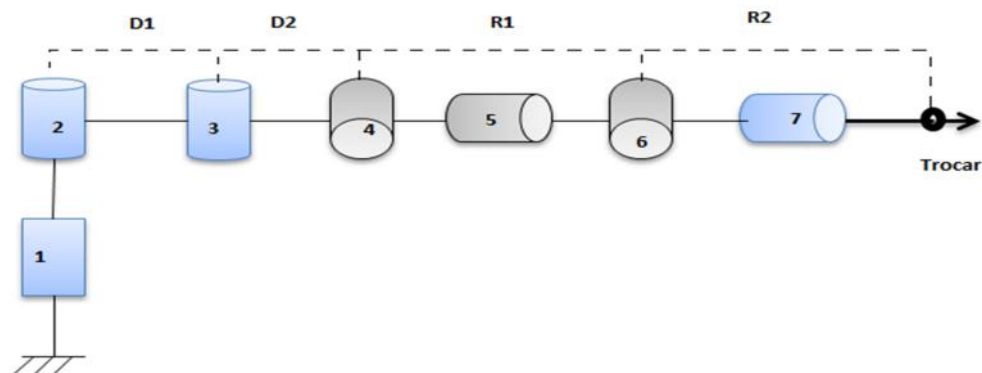


Figura 3.3. Estructura cinemática del robot LapBot rediseñado.

Es importante aclarar que en este proyecto se tomó la decisión de suprimir las dos últimas articulaciones, ubicadas dentro de la cavidad abdominal y que hacían parte del diseño original del LapBot. Esto debido a que dichas articulaciones corresponden a la

orientación de la pinza, la cual no hace parte de los objetivos de este proyecto. Por otro lado las dos articulaciones que no se toman en cuenta son rotacionales y con eje de rotación perpendicular al eje de la articulación anterior (articulación 7). En este caso el movimiento del joystick hace variar la posición en x, y e z del órgano terminal, a través del modelo geométrico inverso, pero no se cambia la orientación del mismo. Suprimir las dos últimas articulaciones no cambia dicha posición. Para futuros trabajos que involucren la pinza como tal será imprescindible añadir dos micromotores que manejen estas dos articulaciones y le den la rotación deseada a la misma.

3.2 Modelado matemático de LapBot

Como ya se había definido la estructura, se podían calcular los modelos matemáticos del robot LapBot para describir analíticamente el movimiento espacial y así verificar que los movimientos quirúrgicos respeten el punto de incisión y no dañen al paciente. Además a partir de estos mismos se parte para realizar el control y simulación del robot. Dichos modelos para el robot LapBot se encuentran en la tesis de maestría "Modelado, Simulación En 3D y Control de un Robot para Cirugía Laparoscópica" [3].

3.3 Diseño general de los cuerpos del LapBot

Los parámetros dinámicos y las dimensiones geométricas del LapBot se definieron teniendo en cuenta aspectos bajo los cuales el proyecto se debía desarrollar, entre ellos:

- La pared abdominal sube entre 0.15 y 0.2m debido al gas de CO₂ que se inyecta al paciente [48].
- La camilla a utilizar tiene un ancho de 70.03 cm, un largo de 103 cm y una altura de 75 cm respecto al piso.
- En el diseño original, el LapBot tenía la base sobre el piso [3], de manera similar al robot para laparoscopia Da Vinci [9]. En el rediseño la base está sujeta a la parte lateral de la mesa, por lo tanto se redimensionaron las distancias entre articulaciones.
- Los motores recomendados en el trabajo "Modelado, Simulación En 3D y Control de un Robot para Cirugía Laparoscópica" [3], Maxon EC Powermax 30 [49], no se utilizaron porque no se pudieron conseguir en el país y la fuente para alimentarlos tampoco era sencilla de conseguir.
- El diámetro del cuerpo que pasa a través del punto de incisión es de entre 0.005 y 0.01 m [50] y [51].

- Para el nuevo diseño se utilizaría un tipo de madera de 1 cm de grosor, que sería cortada por una impresora láser, para lograr la mayor exactitud posible de las piezas.

Partiendo de todo lo anterior para la estructura cinemática del LapBot se obtuvieron las siguientes dimensiones aproximadas:

R2=0.2 m de longitud y 0.008 m de diámetro (cuerpo que se apoya en el trocar). La longitud del cuerpo R1 es de mínimo 0.2 m, este posiciona al robot por encima del abdomen del paciente, D1 y D2 formarían un cuerpo de 0.41 m de largo aproximadamente (Figura 3.3).

El LapBot se ubicaría en una de las partes laterales de la camilla, sujeto a la misma a 0.35 m del centro.

3.3.1 Diseño de las piezas del robot mediante software CAD

El software utilizado para el diseño de las piezas del LapBot fue Solid Edge, considerando que el material escogido era madera de pino (madeflex o triplex) que tenía como espesor 1 cm, además del tamaño y la forma de cada una de las articulaciones y que los motores deberían estar en estructuras que los soportaran de manera segura.

La construcción de cada una de las piezas fue secuencial, empezando desde la base hasta la última estructura que es la que atraviesa el trocar.

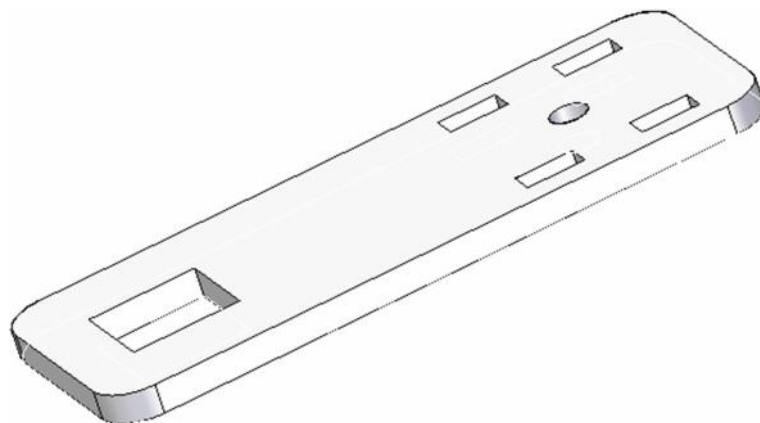


Figura 3.4. Una de las piezas de la articulación dos.

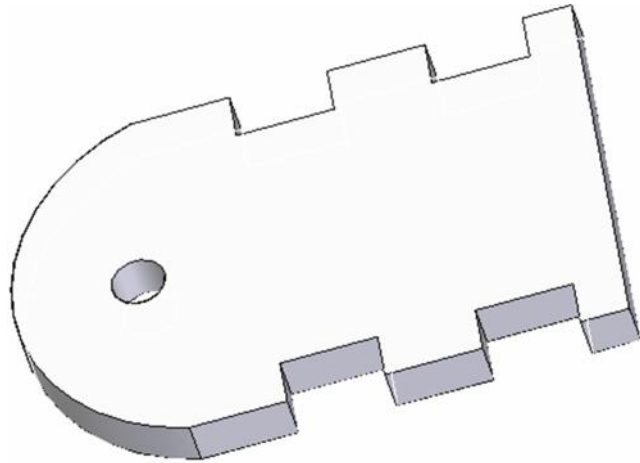


Figura 3.5. Una de las piezas de la articulación tres.

El motor para las articulaciones rotacionales se observa en la figura 3.6. Era necesario simularlo para integrarlo con la estructura de todo el Labor pues su peso afecta considerablemente el movimiento del robot.

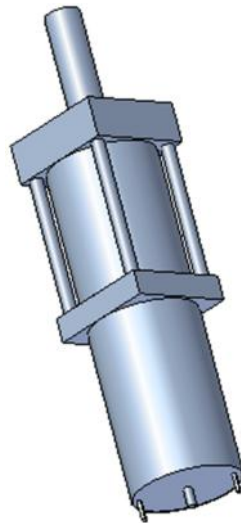


Figura 3.6. Actuador articulación rotacional diseñado en Solid Edge.

La estructura es un conjunto de piezas ensambladas de tal manera que se soportaran los dos motores rotacionales que posibilitaran el movimiento de la segunda y la tercera articulación.

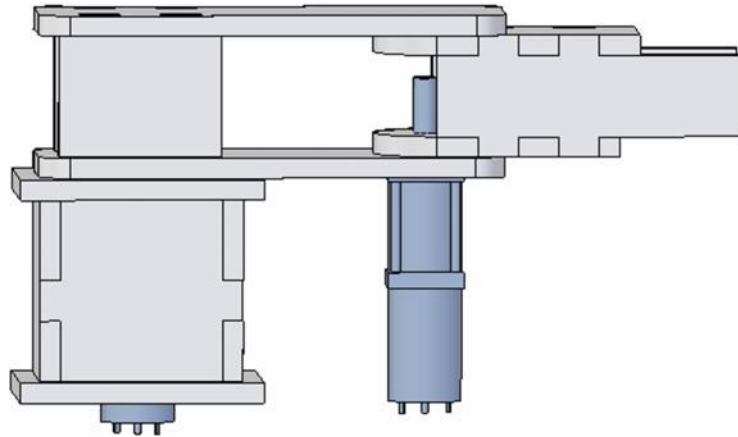


Figura 3.7. Estructura segunda y tercera articulación en Solid Edge.

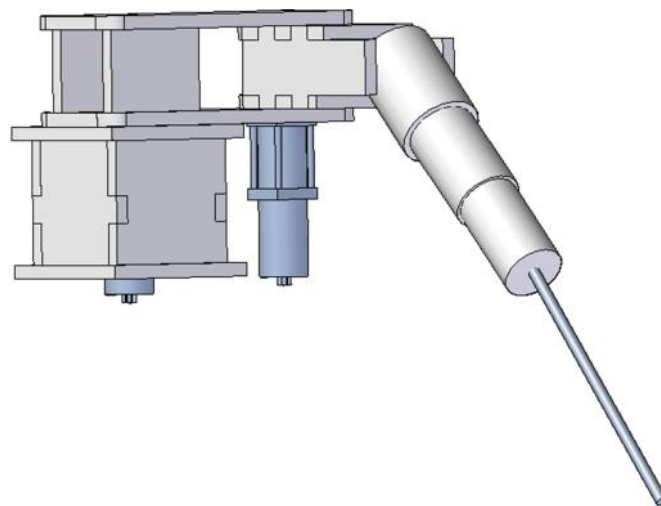


Figura 3.8. Estructura completa del LapBot en Solid Edge.

3.4 Selección de los motores

Para elegir los motores con los cuales se construiría el robot era necesario considerar dos aspectos fundamentales: velocidad y torque del actuador. Una de las principales características del robot Labor es que sus movimientos deben ser lentos, cortos y suaves, es decir, debe manejarse una velocidad baja. Además se debía tener en cuenta que la primera articulación del LapBot es prismática por lo tanto se requería un motor de desplazamiento. Por estos motivos se escogió para la primera articulación un motor que se trasladara y que tuviera la capacidad de levantar bastante peso, ya que este se encargaría de levantar todo el robot, este es el actuador lineal LA31 de la marca Linak[47] que se caracteriza por tener movimientos lentos de acuerdo al voltaje con el que se alimente. Para la segunda y tercera articulación se vio conveniente utilizar motorreductores ya que estos además tenían la ventaja de manejar un alto torque, estos son importados por la empresa Dynamo Electronics.

El último motor estaría ubicado en una de las últimas articulaciones del Labor, la que atraviesa el trocar, cuyo torque no es muy por lo tanto se utilizaría un micro motor que cumpliría con lo requerido.

3.4.1 Cálculo de pares

Para el cálculo de los pares fue necesario obtener los parámetros dinámicos como inercias y masas que conforman el LapBot. Para esto se usó el software CAD SolidEdge que permite agregar tipos de materiales y ejes de rotación a la estructura realizada. Con este programa se pueden obtener parámetros como la masa, el volumen, el centro de masa, la masa de los momentos de inercia y los principales momentos de inercia. Estos datos son introducidos en la simulación en Matlab realizada en el proyecto de grado “Modelado, simulación en 3D y control de un robot para cirugía laparoscópica” [3] y de esta forma se obtienen los pares aproximados de cada articulación.

Para obtener los parámetros dinámicos en SolidEdge primero se debe construir el robot y asignarle a cada pieza y a los motores un material, de esta forma el conjunto tendrá cierta densidad y una vez que se ha hecho esto se deben asignar los ejes de rotación de cada motor. Cuando se han asignado los ejes de rotación se procede a obtener los parámetros de cada articulación, para esto se seleccionan únicamente las piezas que mueven cada motor. Una vez realizado este procedimiento con cada articulación se obtienen los parámetros dinámicos.

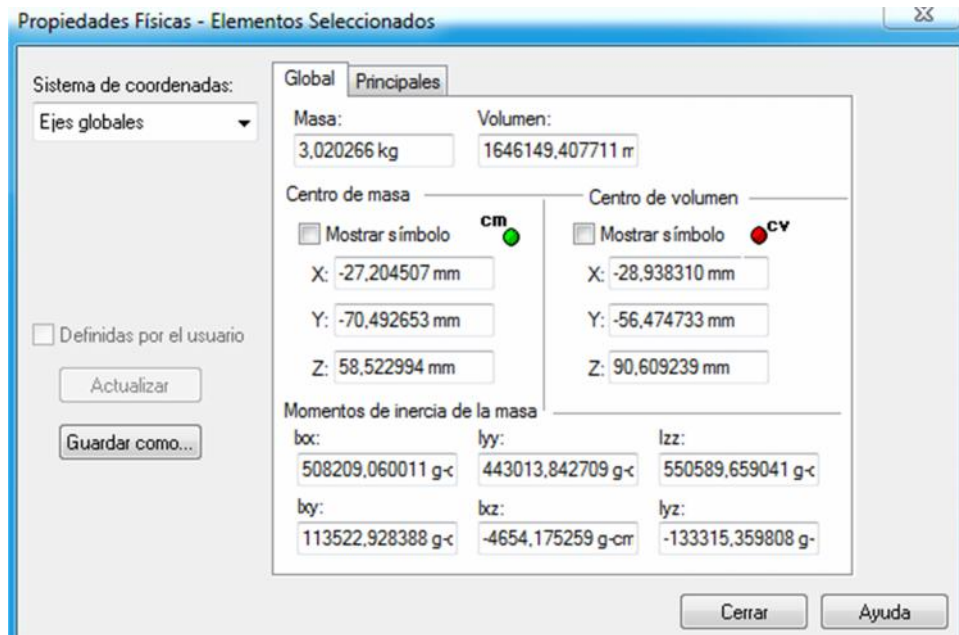


Figura 3.9. Propiedades físicas de la primera articulación obtenidas con SolidEdge.

Articulaciones	Masas de cada articulación
Articulación 1	3,020 kg
Articulación 2	1,760 kg
Articulación 3	0,466 kg
Articulación 4	0,295 kg
Articulación 5	0,199 kg
Articulación 6	0,147 kg
Articulación 7	0,093

Tabla 3.1 Masas de cada articulación obtenidas en SolidEdge.

Todos los valores están en centímetros y kilogramos. Como SolidEdge da algunos datos en milímetros, estos tienen que ser convertidos.

Articulación	Centro de masa x (cm)	Centro de masa y (cm)	Centro de masa z (cm)
Articulación 1	-2,72	-7,049	5,8520
Articulación 2	-4,821	-12,605	7,099
Articulación 3	-18,524	-13,873	6,841
Articulación 4	-26,09	-13,684	2,6390
Articulación 5	-30,854	-13,534	-1,8750
Articulación 6	-34,323	-13,414	-5,2070
Articulación 7	-37,911	-13,237	-8,8000

Tabla 3.2 Distancia a los centros de masa de cada articulación en los ejes X, Y y Z

Para calcular el primer momento de inercia se multiplican las masa de cada articulación por la distancia al centro de masa de cada eje X, Y y Z.

Articulación	Primer momento de inercia en x	Primer momento de inercia en y	Primer momento de inercia en z
Articulación 1	MX1: -8,2144	MY1: -21,28798	MZ1: 17,67304
Articulación 2	MX2: -8,48496	MY2: -22,1848	MZ2: 12,49424
Articulación 3	MX3: -8,632184	MY3: -6,464818	MZ3: 3,187906
Articulación 4	MX4: -7,69655	MY4: -4,03678	MZ4: 0,778505
Articulación 5	MX5: -6,139946	MY5: -2,693266	MZ5: -0,373125
Articulación 6	MX6: -5,045481	MY6: -1,971858	MZ6: -0,765429
Articulación 7	MX7: -3,525723	MY7: -1,231041	MZ7: -0,8184

Tabla 3.3 Primer momento de inercia de cada articulación en los ejes X, Y y Z.

El segundo momento de inercia también lo da SolidEdge pero debe cambiarse a Kg.cm^2 porque viene en g.cm^2 .

Articulación	Segundo momento de inercia en X, Y y Z (Kg.cm^2)		
Articulación 1	XX1: 0,050820906	XY1: 0,011352293	XZ1: -0,000465418
	YY1: 0,044301384	YZ1: -0,013331536	ZZ1: 0,055058966
Articulación 2	XX2: 0,04754864	XY2: 0,011582755	XZ2: -0,000698487
	YY2: 0,04754864	YZ2: -0,014181815	ZZ2: 0,055165219
Articulación 3	XX3: 0,015371744	XY3: 0,011758542	XZ3: -0,000801925
	YY3: 0,029795773	YZ3: -0,004576824	ZZ3: 0,03276291
Articulación 4	XX4: 0,008318919	XY4: 0,010421752	XZ4: 0,000529798
	YY4: 0,025501777	YZ4: -0,001156044	ZZ4: 0,028344917
Articulación 5	XX5: 0,004982306	XY5: 0,008252587	XZ5: 0,002385212

	YY5:0,02152175	YZ5:0,000453893	ZZ5:0,023878933
Articulación 6	XX6:0,003645331	XY6:0,006749099	XZ6:0,003196453
	YY6:0,01889192	YZ6:0,001000201	ZZ6:0,02056692
Articulación 7	XX7:0,002597269	XY7:0,004681118	XZ7:0,003342687
	YY7:0,014610181	YZ7:0,001077909	ZZ7:0,015290896

Tabla 3.4 Segundo momento de inercia en Kg.cm².

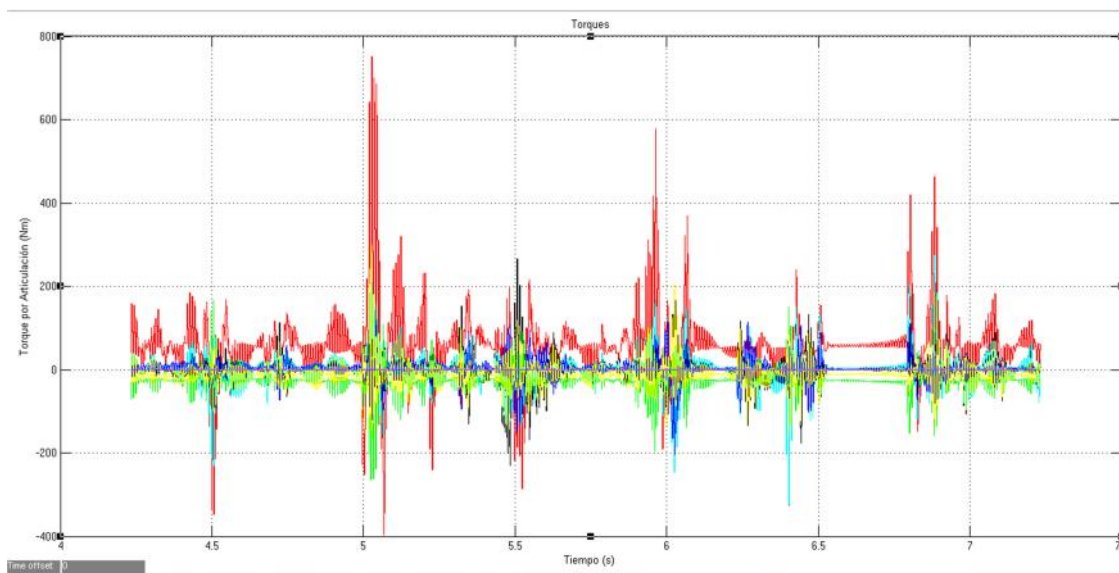


Figura 3.10 Pares obtenidos con Matlab al introducir los parámetros de SolidEdge.

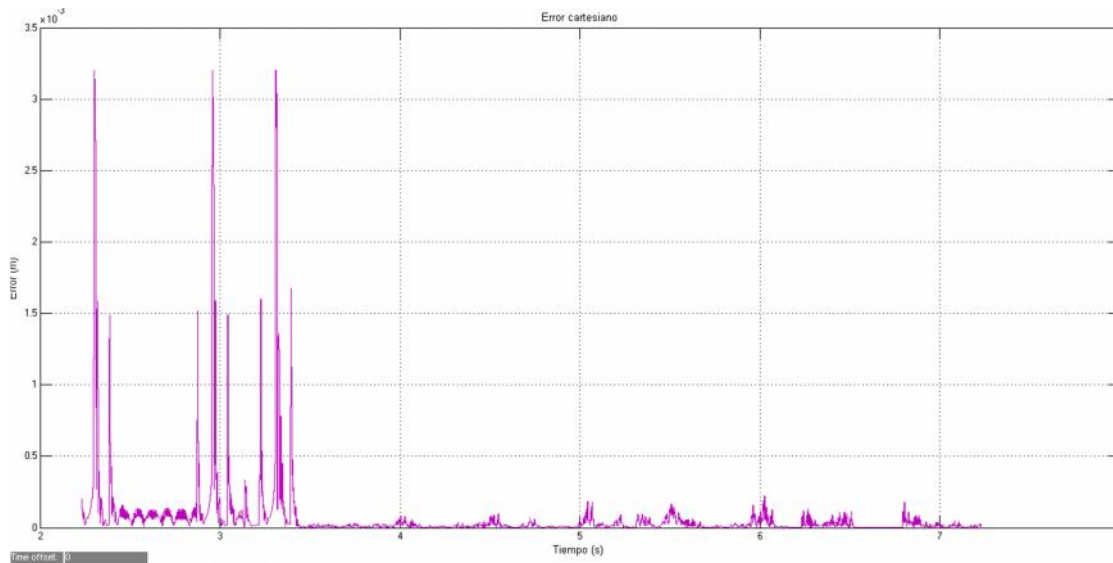


Figura 3.11 Error cartesiano obtenido con Matlab.

La trayectoria que se siguió para la obtención de las gráficas anteriores (figura 3.2 y figura 3.3) fue la de una colecistectomía extractiva, que consiste en quitar una vesícula biliar enferma, que esta inflamada u obstruida por cálculos biliares[3].

Las unidades en que se muestran los torques en Matlab son Newton metro.

Con base en lo obtenido en la Figura 3.2 se escogieron motores que cumplieran con los requerimientos en cuestión de pares. De esta forma se encontraron en el mercado motores rotacionales con 154 kg.cm o 15.1022 Nm de torque y el motor deslizable que resiste una carga de hasta 6000 N.

3.4.1 Rediseño de los cuerpos del LapBot

Se hizo un rediseño de las piezas del robot LapBot, a partir del cual se obtuvieron nuevos valores aproximados de los parámetros de la base y dimensiones geométricas del LapBot.

Fue necesario el rediseño por diferentes razones, entre ellas:

- Los materiales: En un principio se pensó en utilizar acrílico para la construcción del LapBot porque estéticamente se veía mejor debido a su homogeneidad, resistencia y principalmente porque es mucho más liviano en comparación a otros materiales, característica fundamental a la hora de

construir cuerpos que incluyan motores, que por lo general son pesados, y que a su vez realicen algún movimiento.

En cuanto a costos se realizaron las cotizaciones pertinentes en algunos lugares y se eligió la mejor opción en cuanto a precio, calidad y accesibilidad pero debido al elevado costo del material y del proceso de corte no se pudo concretar el uso de este en la construcción y por lo tanto se tuvo que explorar otras alternativas.

Se pensó en utilizar aluminio, pero esta opción fue descartada, porque una de las finalidades era un producto final estético y con el aluminio o metal este objetivo se complicaba pues la forma de algunas piezas era compleja de soldar.

Finalmente se tomó la decisión de construir las piezas en madera y de adaptar el diseño que se tenía a uno nuevo y completamente diferente, porque el material había cambiado y poseía nuevas características. Las piezas debían quedar muy bien acopladas, teniendo en cuenta el tamaño de los motores de cada una de las articulaciones activas, como también la estructura y el mecanismo correspondiente si la articulación es pasiva. La madera tenía nuevas y varias ventajas pues para efectos de pruebas las articulaciones debían ser montadas y desmontadas continuamente, así mismo debían ser instalados diferentes elementos para mantener los motores, las piezas y los potenciómetros en su lugar y para esto se debían utilizar tornillos de diferentes grosores que debían colocarse en varias partes de las piezas, esto se facilitaba en la madera porque perforarla es más sencillo y sufre menos daños.

- Para este proyecto se utilizó como camilla una mesa de madera de 1.01 m de largo por 0.705 m de ancho. En esta se ubican a ambos lados los robots LapBot e Hibou y su distancia máxima es al centro de esta en donde estará ubicado un recipiente que hará las veces de estómago. Los robots deben ir ubicados de tal forma que no se deben tocar.
- El tamaño de los motores fue un factor determinante a la hora de diseñar cada una de las nuevas piezas que conformarían las articulaciones. En el diseño inicial del LapBot[3], se consideraron unos motores que no fue posible conseguir, además que no tenían características compatibles con lo que se buscaba si se quería construir el prototipo. En términos de requerimientos de voltaje y corriente, se dificultaba conseguir una fuente que los hiciera

funcionar adecuadamente, por lo tanto se descartaron. Como el tamaño de dichos motores descartados era considerablemente distinto a los motores con los que finalmente se decidió trabajar, pues estos últimos eran mucho más grandes y pesaban más, las piezas que estos actuadores moverían también cambiarían, sus dimensiones, su forma etc. Por otra parte no se podía descartar la inercia que se podría presentar en los motores elegidos y había que considerar el peso al mover por cada actuador. Los motores elegidos se muestran en las Figuras 3.7 y 3.8.



Figura 3.12. Actuador lineal Linak.



Figura 3.13. Actuador articulación de rotación importado por Dynamo Electronics.

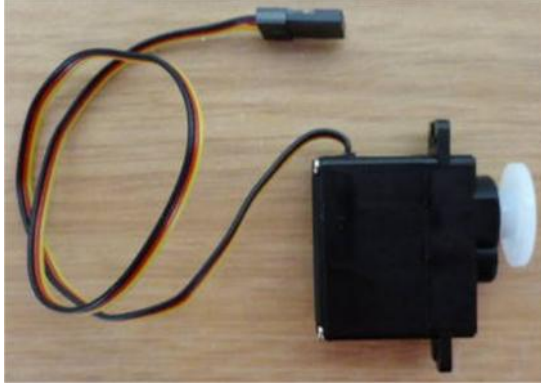


Figura 3.14. Actuador articulación final marca Hitech.

3.4.2 Corte de las piezas del LapBot

Como se dijo anteriormente, los motores fueron analizados con respecto a la forma en que funcionaban en relación al robot, su posición en la estructura y hacia donde se producía el movimiento, cuales articulaciones llevaban motor y en qué orden iban ubicadas, todo esto para prevenir problemas con los movimientos entre articulaciones y demás aspectos relacionados. Esto fue determinante a la hora de formar las piezas y de su posterior corte.

El procedimiento para obtener las piezas consistía en introducir los archivos generados mediante el software CAD, pero tipo AutoCAD, a la impresora láser, la cual cortaba todas las piezas en un determinado tiempo.

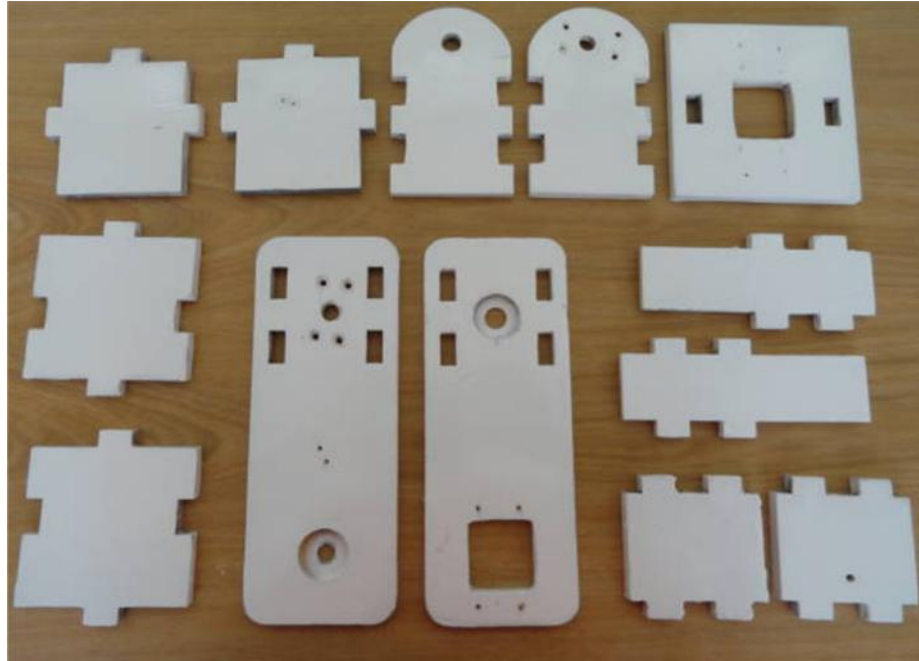


Figura 3.15. Piezas del LapBot



(a)

(b)

(c)

Figura 3.16. Piezas LapBot, cuerpos uno (a), (b) y dos (c).

3.4.4 Articulaciones del LapBot

El prototipo se fue desarrollando secuencialmente partiendo de la base y de su ubicación con respecto a la camilla, siguiendo con la primera articulación luego la segunda y así sucesivamente.

3.4.4.1 Primera articulación

Para el primer grado de libertad, que era una articulación prismática, es decir, de desplazamiento vertical, y que se implementaría con el motor

más grande de todos los que se iba a utilizar para la construcción del LapBot, fue necesario diseñar una base metálica de 18 cmx 16cm de base y 36 cm de alto. Esta se soportaría en la parte lateral e inferior de la mesa o camilla elegida para instalar el LapBot, y en cuyo interior se aseguraría la parte de abajo del motor, de manera que quede fija sin importar la fuerza que se ejerza, garantizando así que el robot no colapse.



Figura 3.17. Distintas vistas de la base metálica del primer actuador.

La parte del motor lineal que sobresale por encima de la mesa atraviesa una placa de metal en la cual se asienta una lámina de madera que será la base para la caja de triplex que la cubrirá y en la cual se pondrá el potenciómetro, las conexiones y un buje de extensión que servirá para conectar esta articulación con la siguiente. Sus dimensiones son de 12.1 cm x 12.5cm de base y 17.1 cm de alto. Esta caja tendría un orificio superior lo suficientemente grande para que el eje del motor pase, esta es la parte que se traslada y que a su vez mueve hacia arriba o abajo el resto del robot.

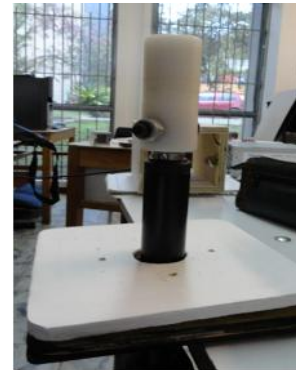


Figura 3.18. Diferentes vistas de la caja contenedora del eje del primer actuador.

El buje de extensión conectado al eje debe tener suficiente resistencia para soportar el peso del resto del motor, es de teflón y está atornillado al eje con un tornillo de seguridad. Arriba tiene un hueco con rosca en donde encaja otro mecanismo que pertenece a la articulación dos.



(a)



(b)

Figura 3.19. Diferentes vistas del buje de extensión (a) y (b).

3.4.4.2 Segunda articulación

Sujeta al buje de extensión de la primera articulación por un tornillo que hace parte de un recipiente que está atornillado a una caja que contiene el segundo motor. La caja es de tamaño suficiente para cubrir el motor pero dejándole espacio. Esta misma se convierte en una base para la segunda articulación, y de allí se desprende el primer cuerpo de movimiento de rotación.



Figura 3.20. Base con caja protectora de la segunda articulación



Figura 3.21. Base de la articulación dos conectada con la articulación uno.



Figura 3.22. Motorreductor de la segunda articulación dentro de la caja



Figura 3.23 Motorreductor protegido completamente por la caja.

Esta segunda articulación consiste en una especie de rectángulo de dimensiones 22.95 cm por 10 cm que rota sobre la segunda caja de madera. Para facilitar la rotación, entre la caja y el rectángulo se ubica una arandela de metal que disminuye el roce. Para facilitar el trabajo del motor y asegurar que al moverse rote toda la

articulación y las piezas siguientes se instala entre el eje del motor y la pared superior de la articulación un adaptador de metal hecho a la medida que se ajusta al eje y se atornilla a la pieza, esto para que se mueva toda la articulación. El adaptador entra a un orificio donde está una balinera y tiene un cilindro hecho para que encaje el potenciómetro.



Figura 3.24. Segunda articulación con los motores.



(a)

(b)

Figura 3.25. Adaptadores hechos a la medida a y b.

Según la figura 3.24 en el extremo izquierdo de esta segunda articulación se ubica un mecanismo que sujeta y asegura al tercer actuador el cual le da movimiento al tercer grado de libertad.

3.4.4.3 Tercera articulación

Es una estructura con un extremo ovalado al cual va conectado el actuador. El otro extremo es rectangular y a este se fija el cuarto grado de libertad. Este es el segundo cuerpo de movimiento de rotación. El motor para este movimiento es igual al del primer cuerpo de movimiento de rotación (grado de libertad anterior) y como ya se mencionó va sujeto a la segunda articulación. La parte ovalada de la pieza se fija al eje del tercer actuador que le brindara el movimiento rotacional.



Figura 3.26. Segunda y tercera articulación.

Esta articulación también tiene una arandela para ayudar a su rotación y un adaptador a la medida para ayudar con su rotación, el adaptador en este caso es un poco más corto pues el eje queda más largo.

3.4.4.4 Cuarta articulación

Al ser una articulación pasiva, no fue necesario hacer de gran tamaño la pieza pues su única función es fijarse a 45 grados para

ubicar el resto de las articulaciones por encima del paciente. Se diseñó como un tornillo que unía la tercera articulación con el quinto y sexto grado de libertad, que eran igualmente pasivos y que mantienen el cuerpo de longitud R2 a través del trocar, manteniendo el centro de rotación remoto. El tornillo mantiene fija una pieza en forma cilíndrica (Figura 3.24) que fue elaborada en PVC y de la cual sale la segunda parte de la quinta articulación.

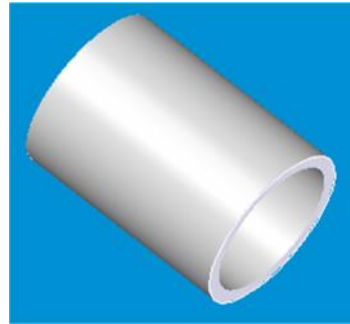


Figura 3.27. Tubo fijo a la cuarta articulación y que permite el inicio de la quinta articulación.

3.4.4.5 Quinta articulación

Esta articulación no tiene motor, por lo tanto debe basarse en un tipo de mecanismo que permita la rotación de un tubo o conducto. Siendo así se decidió utilizar una balinera, porque se debía garantizar que dicha rotación fuera independiente del resto del robot (del cuarto grado de libertad hacia atrás).

Esta articulación se divide en dos partes, ambas en forma cilíndrica, la primera parte del cilindro va fija a la cuarta articulación y no gira, pero la segunda parte del tubo o cilindro es la que se adhiere a la balinera la cual hace que gire el cuerpo, y la articulación cumpla su función.

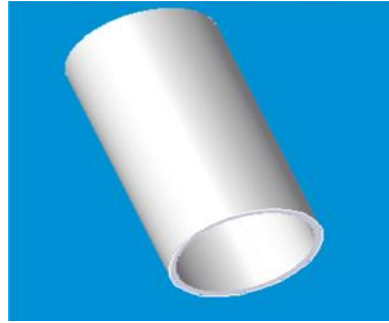


Figura 3.28. Segunda parte de la quinta articulación.

3.4.4.6 Sexta Articulación

Al igual que las dos anteriores, esta articulación también es pasiva. Su función es otorgar al elemento final del LapBot, estructura que atraviesa el trocar, la capacidad de moverse hacia arriba y abajo, aun lado y al otro, en un rango limitado, manteniendo el punto de incisión fijo. A su vez va sujeta a la quinta articulación, por lo tanto el efecto de la balinera del quinto grado de libertad recae igualmente sobre ella, permitiendo que también gire.

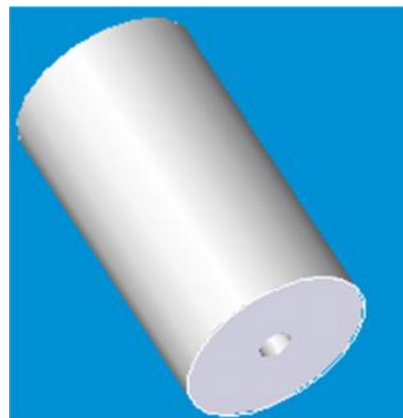


Figura 3.29. Sexta articulación.

3.4.4.7 Séptima articulación

Esta última articulación a implementar en la estructura es activa. El motor escogido para el movimiento de la misma es un micromotor

pues es una articulación pequeña que no pesa mucho ni tiene mucho torque. En este grado de libertad, el elemento final gira dentro del trocar, manteniéndolo como punto de referencia. El efecto de las dos articulaciones anteriores también se transmite por esta estructura.



Figura 3.30. Séptima articulación.



(a)



(b)



(c)

Figura 3.31. Distintas vistas de las articulaciones 4, 5, 6 y 7. A, b y c

3.5 Selección de los potenciómetros

Los potenciómetros son los encargados de medir la posición de cada una de las articulaciones del LapBot.

En primer lugar para la articulación prismática se necesitarían un sensor lineal deslizable, para relacionar el desplazamiento y el voltaje (figura 3.24). Este potenciómetro se ubicaría en la parte lateral del primer motor paralelo al eje y es de 20kiloohmios. Para las articulaciones segunda, tercera y séptima, se eligieron potenciómetros igualmente lineales pero convencionales, con los cuales se relacionaría la posición angular y el voltaje, y que se ubicarían en el eje del motor respectivo. El potenciómetro utilizado es un Vishay 533 de 10 kiloohmios a 3 vueltas, en su hoja de datos el fabricante especifica que es lineal (figura 3.25). Los potenciómetros seleccionados deben generar valores de voltaje que tengan una relación directamente proporcional con la cantidad de centímetros desplazados (articulación prismática) o cantidad de grados en los ángulos resultantes después de un desplazamiento rotacional, dependiendo de la articulación que sea. Para verificar lo antes dicho se realizaron pruebas con los potenciómetros polarizándolos con un valor fijo de voltaje variando el desplazamiento bien sea en centímetros o en grados, según el tipo de potenciómetro.



Figura 3.32. Potenciómetro deslizable.



Figura 3.33. Potenciómetro Vishay.

Para comprobar la linealidad de un potenciómetro deslizable de 20K, por ejemplo se tomaron muestras como se observa en la tabla 3.5.

Distancia (mm)	Resistencia (K)
5	0,41
10	0,97
15	1,57
20	4,87
25	7,81
30	11,31
35	14,31

40	17,20
45	19,28
50	19,87
55	20,3
60	20,4

Tabla 3.5 Medición potenciómetro deslizable (distancia – resistencia).

Para encontrar el intervalo en que el potenciómetro deslizable se comporta linealmente, se toma la relación entre la distancia y la resistencia y de allí se obtiene dicho intervalo que está entre los 15 y 40mm.

Ahora se debe encontrar la relación entre la distancia recorrida por el potenciómetro y el voltaje correspondiente a cada distancia, estos resultados están contenidos en la tabla 3.6.

Distancia (mm)	Voltaje (Voltios)
5	0,06
10	0,15
15	0,27
20	0,79
25	1,28
30	1,84
35	2,38
40	2,87
45	3,16
50	3,25

55	3,34
60	3,38

Tabla 3.6. Medición potenciómetro deslizable con voltaje (distancia – voltaje).

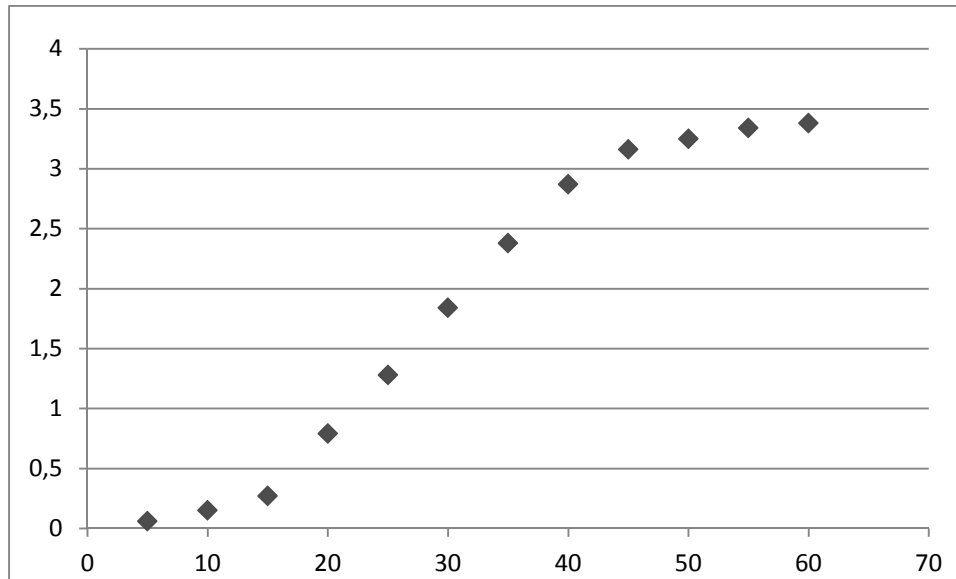


Figura 3.34. Relación voltaje - distancia.

Por otro lado está el potenciómetro que relaciona desplazamiento angular, que también es lineal, de este mismo se obtuvieron las muestras de la tabla 3.7.

Ángulo (grados)	Voltaje
10	0,1158
20	0,231
30	0,35
40	0,463
50	0,579

60	0,688
70	0,81
80	0,931
90	1,05
100	1,16
110	1,27

Tabla 3.7. Medición potenciómetro con voltaje (ángulo- voltaje).

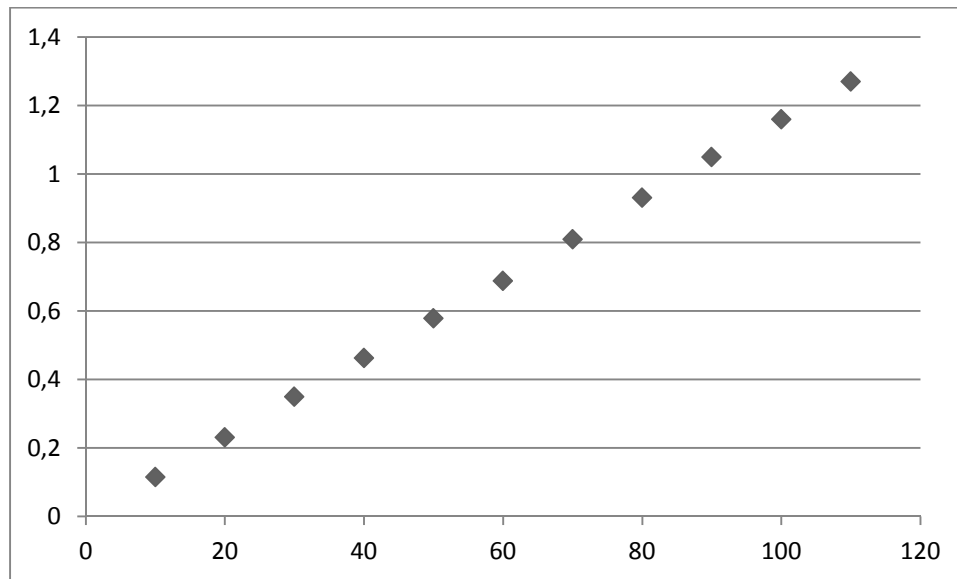


Figura 3.35. Relación voltaje - ángulo.

3.6 Selección del puente H

Después de escoger los motores era necesario poder moverlos en ambos sentidos, tanto para avance como para retroceso, labor de la que estaría encargado un puente H. El puente H seleccionado debería cubrir los requerimientos de corriente y potencia, por lo tanto una buena opción era el puente H Pololu *high-power motor driver* 18V 15A. Esta tarjeta tiene un puente H formado por MOSFET que puede entregar de 5.5 a 30 voltios y 15 amperios (figura 3.31).

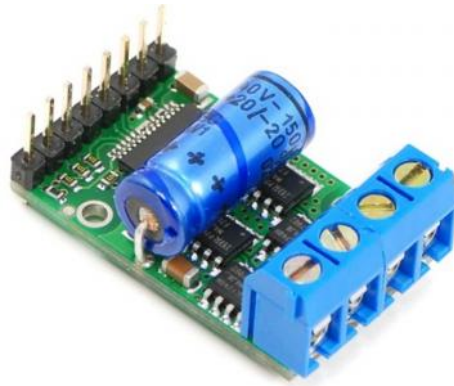


Figura 3.36. Puente H Pololu high-power motor driver 18v15.

3.7 Fuente de voltaje

La fuente seleccionada debe cumplir con los requerimientos en cuanto voltaje y corriente para toda la estructura del LapBot. Debe tener suficiente corriente para alimentar todos los motores del robot. La fuente seleccionada, que se observa en la figura 3.32 es una fuente utilizada en los computadores y ofrece un valor de corriente de 31 amperios a 5 voltios, suficiente para que el LapBot funcione, otorgando además protección por si se da el caso de una sobrecarga.



Figura 3.37. Fuente de alimentación.

3.8 Tarjeta de adquisición de datos

La tarjeta de adquisición de datos debía cumplir como mínimo con los siguientes requerimientos:

- Conexión mediante protocolo de comunicación USB.
- Capacidad de utilizar 8 conversores análogos-digitales.
- Bajo consumo de energía.
- Capacidad de utilizar salidas digitales para manipulación de los motores
- Que se pueda integrar a la plataforma de desarrollo.
- Entorno de desarrollo conocido o de fácil aprendizaje.

Teniendo en cuenta los requerimientos anteriores se eligió un microcontrolador de Microchip por su fácil manipulación, su conexión sencilla y porque la programación no sería muy difícil. El microcontrolador seleccionado para la tarjeta fue el 18F4550 (figura 3.28) porque se adaptaba adecuadamente a los requerimientos soportando comunicación USB, con entradas y salidas suficientes para todos los motores. Otros elementos como el cristal de cuarzo, capacitores, resistencias y un conector USB, fueron igualmente incluidos en la tarjeta, para que así el micro controlador pudiera ejecutar sus funciones a cabalidad.



Figura 3.38. Microcontrolador 18F4550.

Posteriormente se realizó el diseño de la tarjeta en circuito impreso, integrando el microcontrolador y los demás elementos necesarios.

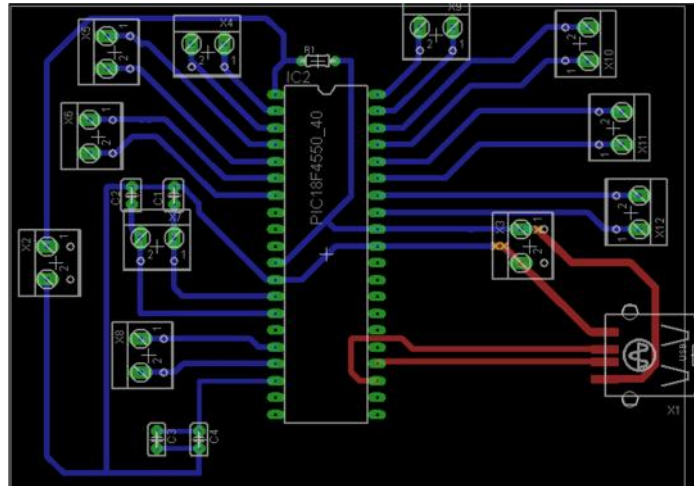


Figura 3.39. Circuito impreso de la tarjeta de adquisición de datos.

La tarjeta de adquisición de datos se encarga de la comunicación entre el robot y el computador, esto se hace a través del puerto usb a través del cual se envían y se reciben las señales.

Cada tarjeta de cada robot tiene un número de identificación diferente, lo que le permite funcionar sin entrar en conflicto con la otra.

La forma de adaptar estos números de identificación para evitar conflictos se explicará en los siguientes capítulos.

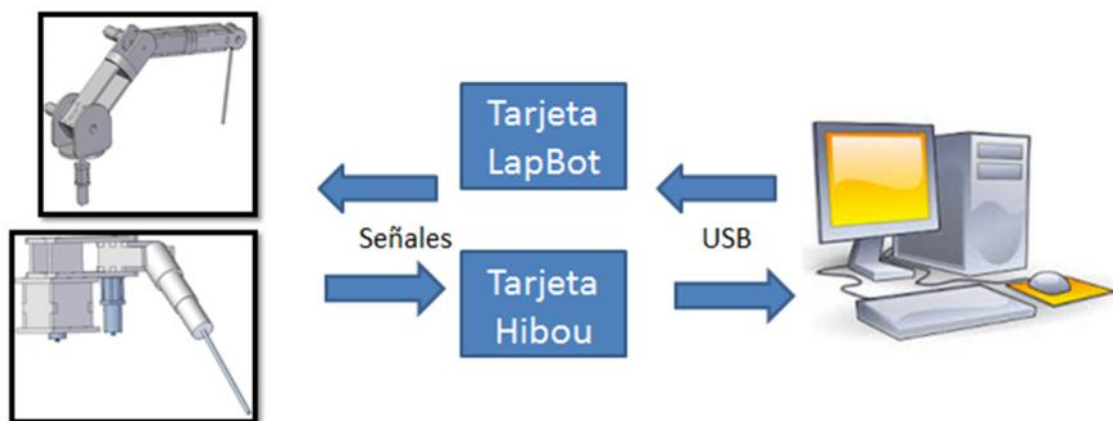


Figura 3.40. Esquema de la comunicación entre el computador y el robot.

4. Software para la manipulación del Lapbot

Para manipular el Lapbot se utilizan dos tipos de software uno es el software del ambiente virtual creado en la asignatura “Aplicaciones de la robótica médica” de la maestría en Automática Industrial por los ingenieros Luis Daladier Guerrero y Cristian Méndez en la Universidad del Cauca y el otro es el software desarrollado para el micro controlador en el programa MPLab.

Para realizar dicho software se utilizaron las siguientes librerías y plataformas:

- Visual Studio (plataforma de desarrollo).
- Librería VTK (motor gráfico).
- MPLab (plataforma de desarrollo para micro controladores)

4.1 Adaptación del software para el Lapbot

Para que se pudiera manipular el Lapbot era necesario adaptar el código del software ya existente, de manera que este soportara la comunicación USB entre el computador y el microcontrolador. Para dicho fin se realizaron diferentes acciones entre ellas la inclusión de nuevas librerías, declaración de nuevas variables y modificaciones en el Pic entre otras.

También se adaptó el código para que funcionara con dos joystick, uno para cada robot y así poder moverlos al mismo tiempo. En el código original los robots se manipulaban con un solo joystick, lo que impedía la manipulación de los dos al mismo tiempo y no permitía que trabajaran en conjunto.

4.1.1 Integración de la librería MPUSBAPI.DLL al software para la manipulación

Para enviar los datos desde el computador hacia la tarjeta de adquisición, se debe integrar la librería MPUSBAPI al software de manipulación del robot. Los datos a enviar son las posiciones articuladas deseadas que se calculan por medio del modelo geométrico inverso del robot LapBot implementado en el software.

4.1.2 Vinculación de MPUSBAPI.DLL a Visual Studio 2008

Para implementar una conexión USB para manipular el robot LapBot, al software existente, se le debe agregar una librería para la conexión USB. Esta librería es la MPUSBAPI.DLL. Para lograr esta integración se deben agregar a la carpeta donde se encuentra el proyecto tres de los cuatro archivos que conforman la librería, estos son: usb4550.h, usb4550.lib y stdafx.h.

4.1.3 Declaración de variables locales para utilizar la librería MPUSBAPI.DLL

- PCHAR vid_pid = "vid_xxxx&pid_yyyy"; es la cadena de caracteres del número de identificación asignado del puerto USB, donde xxxx y yyyy son los valores hexadecimales del VID y el PID respectivamente.

Para el caso del microcontrolador del LapBot su VID es 0x04d8 y el PID es 0013, entonces la cadena de entrada es: "vid_04d8&pid_0013".

Esta cadena tiene que ser igual en el código de Visual y en el código de MPlab, pues es una de las condiciones para que los dos códigos mantengan la conexión interna que permitirá que los movimientos del LapBot virtual se puedan replicar en el real a través del microcontrolador.

- Los *pipes* son los que realizan la comunicación entre el cliente y el dispositivo USB. El cliente software USB debe abrir *pipes* de la misma forma que se abren archivos. Varios *pipes* de un solo dispositivo pueden ser abiertos al mismo tiempo. Los *pipes* de USB son abiertos por los *endpoints*. Los descriptores *endpoints* contienen información sobre la dirección de las entradas y salidas, control de transferencia y el tamaño máximo de paquete admitido [52].
 - PCHAR out_pipe= "\\MCHP_EP1"; y PCHAR in_pipe= "\\MCHP_EP1"; son cadenas de caracteres con el número del *endpoint* que se va a abrir para establecer la comunicación. El formato es \\MCHP_EPz, donde z es el número del *endpoint*. En este caso out_pipe es el *endpoint* 1 para escribir datos en el puerto USB e in_pipe es el *endpoint* 1 para leer los datos del puerto USB.
 - HANDLE myOutPipe; Es la variable que identifica el *pipe* del *endpoint* que se va a escribir.
 - HANDLE myInPipe; Es la variable que identifica el *pipe* del *endpoint* que se va a leer.
 - BYTE SendData[x]; variable donde se almacenan los datos que se envían por el puerto USB.
 - BYTE ReceiveData[x]; variable donde se almacenan los datos que se reciben por el puerto USB.

- DWORD SendLength = x; número de bytes que se van a escribir en el *pipe*.
- DWORD ExpectedReceiveLength = x; número de bytes que se van a leer en el *pipe*.
- DWORD SentDataLength; puntero al número de bytes que se escriben.
- DWORD ReceiveLength; puntero al número de bytes que se leen.
- DWORD SendDelay = 5; especifica el intervalo de *time-out* en milisegundos.
- DWORD ReceiveDelay = 10; especifica el intervalo de *time-out* en milisegundos.

4.1.4 Funciones de la librería MPUSBAPI.DLL

- **MPUSBOpen:** Función que devuelve el acceso al *pipe* del *endpoint* con el VID_PID asignado.
- **MPUSBRead:** Función que permite leer los datos del Puerto USB.
- **MPUSBWrite:** Función que permite enviar los datos en el Puerto USB.
- **MPUSBClose:** Función que cierra la comunicación.

4.2 Software del microcontrolador

El software para lograr la manipulación del robot a través de un joystick y del puerto USB se desarrolla en MPLab y es de tipo On-Off.

Este código se desarrolla teniendo en cuenta las características de los potenciómetros como el número de vueltas y el voltaje con el que se van a alimentar. Esto para tener mejores datos que entren por el conversor análogo digital.

Para poder comparar los valores deseados con los valores obtenidos todos deben ser convertidos a bytes.

El código manipula los motores basado en la posición deseada, esta se da a través del código en Visual. La posición deseada se guarda en una variable que se utiliza para compararla con el valor que está devolviendo el potenciómetro y que se lee a través del conversor analógico digital. Cuando el valor del potenciómetro es menor que el valor de la posición deseada el motor se prende y avanza o gira despendiendo de la articulación hasta alcanzar el valor deseado, es aquí cuando el motor se apaga y se mantiene en esa posición hasta que se le da una nueva orden desde el computador. Y como se espera, si el valor que se está leyendo es mayor que el deseado el movimiento será inverso.

5. Pruebas y resultados

Para poder obtener resultados válidos con el robot construido en su totalidad se tuvieron que realizar una gran cantidad de pruebas a los diferentes elementos que lo componen, así como pruebas de hardware y de software y varias modificaciones a las piezas que lo componen.

5.1 Pruebas hardware

Las primeras pruebas del hardware se realizaron con los motores, estos fueron probados con diferentes valores de voltaje para verificar su velocidad e inercia. Dichas pruebas hicieron para cada motor y de dos formas distintas, para las primeras se conectó cada uno de los motores directamente a la fuente y en las segundas pruebas se conectaba cada motor al puente H. Con las pruebas realizadas a los motores, conectándolos directamente a la fuente se descubrió que la velocidad ideal para los motores rotacionales se daba con la alimentación de 5V pues apenas se apagaba la fuente el motor se detenía. Por otro lado si se alimentaba con 12V, que es el voltaje recomendado por el fabricante de los motores, se movían mucho más rápido, afectando la inercia de los motores, porque al apagar la fuente estos seguían en movimiento. Con las pruebas realizadas con el puente H se buscó entender la función de los diferentes pines, especialmente los de dirección y el de PWM. El pin de dirección como su nombre lo indica da la orden al motor de girar o desplazarse a un lado o a otro, esto se hace si se le envía un cero o tierra, o un uno o 5V. El pin de PWM se debe alimentar con 5V para que el puente H funcione y permita el movimiento del motor, si esto no se hace no habrá ningún movimiento. Cabe resaltar que los pines mencionados no deben ser alimentados con más de 5V o se quemará el puente H.

Con el motor lineal se dieron resultados diferentes pues el voltaje recomendado por el fabricante es de 24V, por lo tanto con 5V no se movía pero con 12V tenía una velocidad ideal para lo que se requería, pues hay que tener en cuenta que esta es la base para

un prototipo de robot quirúrgico y estos robots se caracterizan por tener movimientos muy cuidadosos.

Las pruebas se realizaron sin carga por lo que no podían tomarse como definitivas pues los motores iban a tener que mover mucho peso.

Las pruebas del puente H también se realizaron sin carga, de esta forma no generó problemas para manejar el sentido de los motores, pero una vez más era necesario probar que daba la potencia necesaria cuando los motores tuvieran una carga.

El primer motor en ser probado con carga fue el actuador lineal Linak. A este motor se le adaptó un asiento sobre el cual un hombre promedio de aproximadamente 60 kg le hizo fuerza hacia abajo mientras el motor subía. El motor no tuvo ningún problema en levantar su peso y el amperaje máximo que se dio ante tal esfuerzo fue de 3 amperios siendo que el motor soporta hasta 5.5 amperios.

Los motores rotacionales se probaron con carga cuando se les fueron instalando las piezas de cada articulación. Cuando al primer motor rotacional, correspondiente a la segunda articulación, se le instalaron las piezas y se le agregaron varios tornillos que lo ajustaron ya no se pudo mover con 5V por lo tanto ese valor de voltaje fue descartado pero se mantenía la duda de la inercia generada por la velocidad del motor a los 12V. Sin embargo una vez realizadas las pruebas se constató que el peso de las piezas y los otros elementos, más el ajuste de los múltiples tornillos permitieron que el motor trabajara bien y no generara inercia.

Con el segundo motor rotacional sucedió algo similar que con el primero pues no fue sino hasta que se instalaron todas las piezas y se ajustaron los tornillos que se verificó que se movía con 12V, sin embargo esta pieza sola es muy liviana y sí genera inercia, pero después de varias pruebas se verificó que esta pieza al ser más liviana que la anterior si se movía con 5V y así no tenía inercia.

Para probar con la carga original al motor lineal se le tuvieron que adaptar varias piezas para que se conectara con el resto de las articulaciones y para que se mantuviera estable en la mesa, sin embargo al energizarlo se notó que el motor no solo se desplazaba hacia arriba como se requería sino que también giraba y el giro se hacía más notable con las piezas del LapBot instaladas. Para corregir esto primero se verificó que el giro no fuera fundamental en el funcionamiento del motor, una vez verificado esto se buscaron diferentes soluciones entre las que se destacó instalar un alambre grueso (figura 5.2) y con este crear un redondel por el cual entra la palanca del potenciómetro deslizante, de esta forma al girar el motor el potenciómetro lo detiene ya que el giro no es fuerte y de paso se aprovecha que el redondel mueve el potenciómetro para medir su movimiento. Con esto se solucionó el problema del giro y el motor lineal se movió satisfactoriamente.

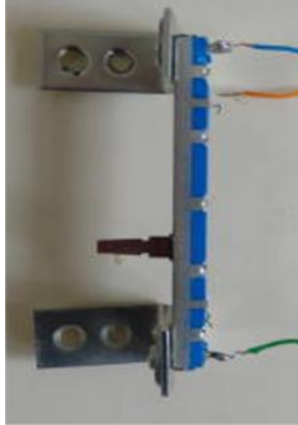


Figura 5.1 Potenciómetro deslizable con eles instaladas para evitar el giro del primer actuador



Figura 5.2 Buje adaptado para que funcione con el potenciómetro deslizable

Los puentes H también fueron probados con carga y no presentaron ningún inconveniente con la potencia.

Otros elementos que se probaron fueron los potenciómetros tanto el deslizable como el convencional. Las primeras pruebas fueron para probar linealidad, estos resultados están consignados en las tablas 3.5, 3.6 y 3.7. Para los potenciómetros Vishay 533 se debió tener en cuenta un rango especial pues aunque son de 10 kilohmios dan 3 vueltas, algo que el robot no puede alcanzar así que se decidió aplicarle un voltaje de 12V y dividirlo en tres vueltas lo que permite hacer cálculos para el código del microcontrolador con 4V por vuelta.

Sin embargo, al hacer la regla de tres que relaciona la cantidad de grados que cambian al moverse el motor con el voltaje y la resolución del microcontrolador, dio como resultado que por cada grado que se variaba no se leían suficientes bytes. Debido a esto se decidió alimentar los potenciómetros de los motores rotacionales con una fuente de 24V, esto hizo que la resolución aumentara y de esta forma se mejorara la lectura de los datos.

La tarjeta de adquisición de datos se probó con los softwares. Primero se hicieron pruebas con leds para verificar que el microcontrolador estaba enviando y recibiendo señales y así se comprobó también que no estaba dañado y que las conexiones de la tarjeta estaban bien hechas.

5.2 Pruebas software

Las pruebas software se realizaron como se dijo en el punto anterior con el microcontrolador. Lo primero fue verificar que la conexión USB estaba funcionando. Para esto se instaló un controlador del USB en el Pic y se verificaron los números vid y pid para que coincidieran en ambos códigos. De esta forma iban a estar conectados entre sí y el USB iba a tener un controlador como el de cualquier otro puerto. Para verificar el controlador del microcontrolador solo se debe conectar al computador y comprobar que suene la alarma que tienen todos los computadores cuando se conecta cualquier otro dispositivo USB y para confirmar se revisa su existencia en el controlador de dispositivos del equipo.

Una vez realizado esto, se probó la comunicación USB utilizando el joystick, para esto se utilizaron dos leds que representaron las salidas que le dan al motor la velocidad y la dirección. Una vez abierto el programa se movía el joystick y se comprobaba que los leds reaccionaban a las órdenes.

Otra prueba que se realizó fue verificar el tipo de movimientos que realizaba el software con el joystick. Para esto se hicieron cuatro movimientos básicos: adelante, atrás, izquierda y derecha. Se verificó con esto que los movimientos del joystick generaban una reacción inversa en algunas articulaciones en el software, esto con el fin de asegurar que la articulación siete se mueva igual que el joystick, así si se realiza por ejemplo un movimiento hacia adelante la articulación siete se va hacia adelante aunque las otras articulaciones no lo hagan.

5.3 Pruebas conjuntas

Después de verificar el funcionamiento del hardware y el software por separado se procedió a realizar pruebas de todo el sistema unificado.

Primero se probó cada motor con su respectivo puente H para comprobar que realizaban algún movimiento al enviar señales desde el software. Esto se hizo solo con una parte del software pues al ser la primera prueba con motores, no estaban aún instalados los potenciómetros.

Luego se realizaron las pruebas de los motores con los potenciómetros. Con estas pruebas había que tener en cuenta que los potenciómetros debían estar en un valor fijo al empezar el movimiento, este valor para los potenciómetros de 10 kiloohmios es de aproximadamente 1 Kiloohmio. La razón para escoger este valor como punto inicial tiene que ver con la característica del potenciómetro de tener su valor máximo al dar tres vueltas, estas tres vueltas en grados serían 1080 pero la resolución del conversor analógico digital es solo de 1024, además se debía tener en cuenta el voltaje con el que se alimentaría el potenciómetro. Se escogió un voltaje de 24V, de esta forma se obtendrían más grados de referencia. Para obtener el valor de referencia en ohmios se realizó una regla de tres. Como no se tenía un voltaje de referencia se tomó el valor de 5V que es el valor máximo del conversor análogo digital y que se convierte a 1024, esto para tener la seguridad de no sobrepasar el valor de la resolución. Así, si 24V equivalen a 10 kiloohmios entonces 5V equivalen a 2.08 kiloohmios, de este último valor se toma la mitad para tener la libertad de mover los motores de un lado a otro sin preocupaciones de sobrepasar los valores máximos del conversor. Si el potenciómetro no está ubicado cerca del valor inicial, el robot se moverá buscando ese valor, cuando lo encuentre se quedará quieto.

Con el potenciómetro deslizable fue más sencillo trabajar pues este se mueve solo seis centímetros y el rango de trabajo está claramente limitado por el rango donde es lineal, como se ve en la tabla 3.6. Esto permite que la alimentación del potenciómetro deslizable sea de 5V.

Una vez realizadas estas pruebas se procedió a probar el funcionamiento del robot ya montado, esto es con todas las articulaciones conectadas entre sí.

La primera prueba se realizó sin el software, conectando los motores, el puente H y la fuente para verificar que el movimiento de cada motor no afectara la posición de otras articulaciones.

Una vez realizadas las pruebas sin el software, este fue conectado. Primero se probaron cada uno de los motores por separado para observar como su movimiento

afectaba al resto del sistema. Una vez que se había probado cada motor individualmente se procedió a probar el movimiento de todo el conjunto, estas pruebas tuvieron que hacerse con mucho cuidado pues un mal movimiento podría ocasionar daños en la estructura del robot. Se verificó su funcionamiento con 4 posiciones básicas: adelante, atrás, derecha e izquierda. Estas pruebas se muestran en las figuras 5.3 a 5.8, las cuales incluyen la posición inicial. En estas figuras el LapBot se muestra a la derecha y el Hibou a la izquierda.



Figura 5.3 Sistema completo.

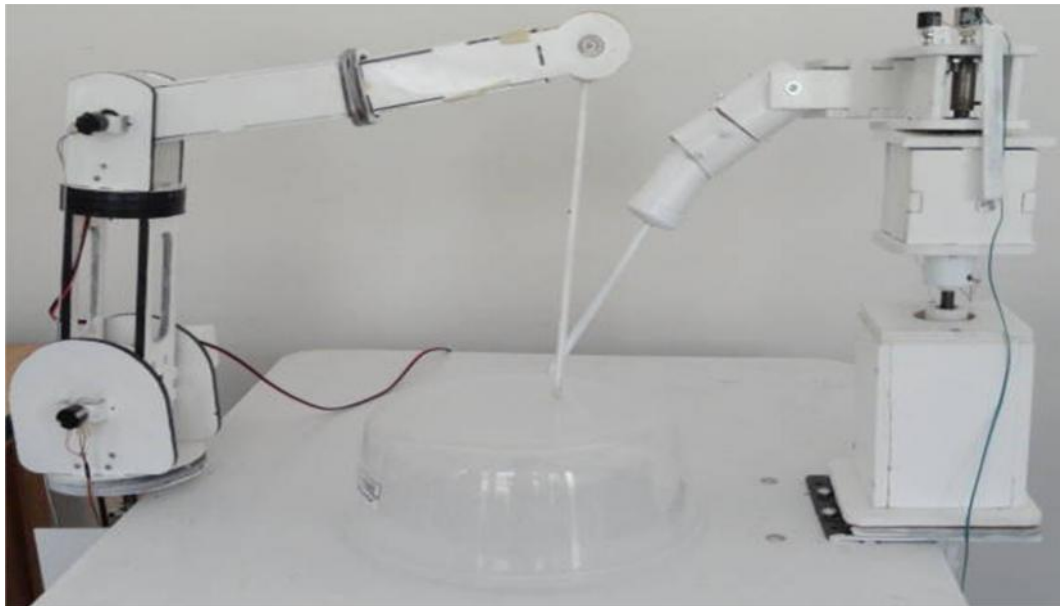


Figura 5.4 Posición inicial del LapBot



Figura 5.5 Movimiento hacia adelante del LapBot.

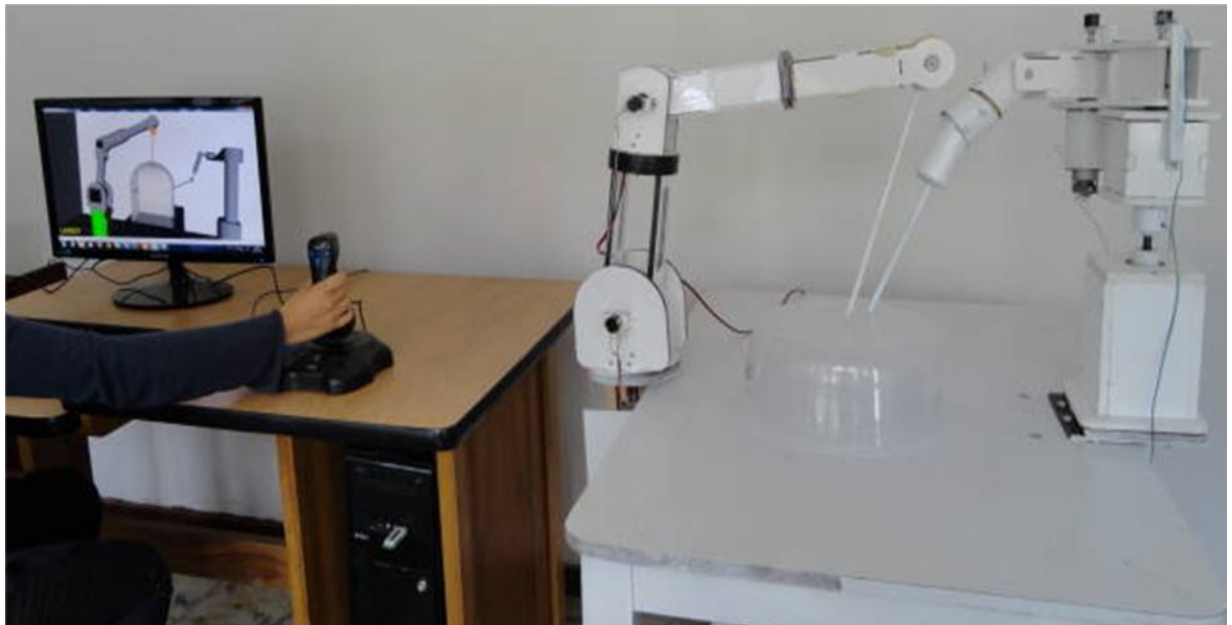


Figura 5.6 Movimiento hacia atrás del LapBot.



Figura 5.7 Movimiento hacia la izquierda del LapBot.

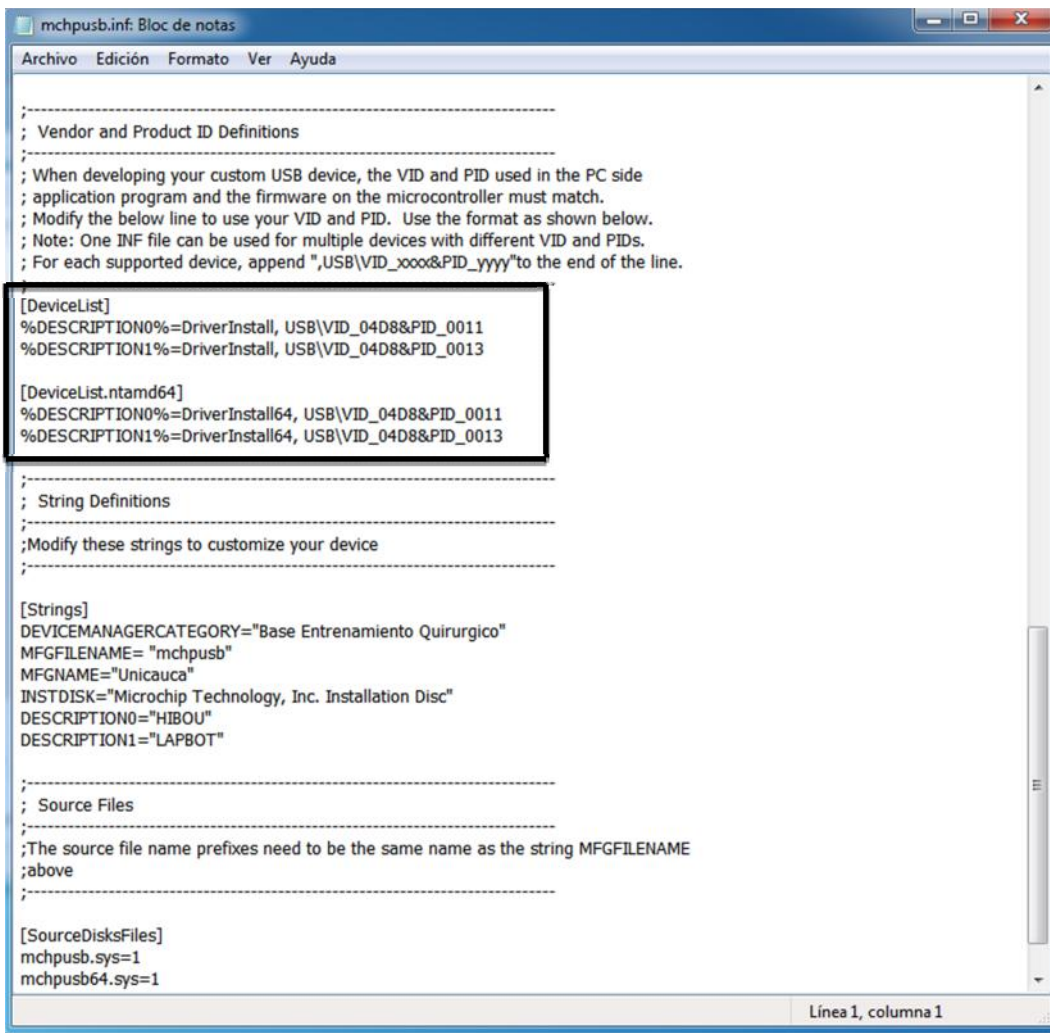


Figura 5.8 Movimiento hacia la derecha del LapBot.

5.4 Integración de los robots Hibou y Lapbot.

Después de verificar que cada uno de los robots funcionaran independientemente, se procede a realizar las modificaciones necesarias para que pudieran trabajar en

conjunto, es decir, que ambos se manejan desde un mismo computador y fueran manipulados por sus joysticks respectivos. Para dicha integración fue necesario editar el driver de las tarjetas de cada robot en el computador principal, para esto se debió crear un solo driver que tuviera en un solo archivo los dos valores de VID Y PID de los dos dispositivos a conectar. Al ser diferentes estos valores de identificación de las tarjetas se evitan conflictos de lectura y al ser un solo archivo se pueden manipular los dos robots al tiempo. En la figura 5.9 se puede observar el reconocimiento de ambas tarjetas, con sus identificadores respectivos, además se puede apreciar el valor y la descripción que se le asignan a cada una.



```
mchpusb.inf: Bloc de notas
Archivo Edición Formato Ver Ayuda

;-----
; Vendor and Product ID Definitions
;-----
; When developing your custom USB device, the VID and PID used in the PC side
; application program and the firmware on the microcontroller must match.
; Modify the below line to use your VID and PID. Use the format as shown below.
; Note: One INF file can be used for multiple devices with different VID and PIDs.
; For each supported device, append ",USB\VID_XXXX&PID_YYYY" to the end of the line.

[DeviceList]
%DESCRIPTION0%=DriverInstall, USB\VID_04D8&PID_0011
%DESCRIPTION1%=DriverInstall, USB\VID_04D8&PID_0013

[DeviceList.ntamd64]
%DESCRIPTION0%=DriverInstall64, USB\VID_04D8&PID_0011
%DESCRIPTION1%=DriverInstall64, USB\VID_04D8&PID_0013

;-----
; String Definitions
;-----
;Modify these strings to customize your device
;-----

[Strings]
DEVICEMANAGERCATEGORY="Base Entrenamiento Quirurgico"
MFGFILENAME= "mchpusb"
MFGNAME="Unicauca"
INSTDISK="Microchip Technology, Inc. Installation Disc"
DESCRIPTION0="HIBOU"
DESCRIPTION1="LAPBOT"

;-----
; Source Files
;-----
;The source file name prefixes need to be the same name as the string MFGFILENAME
;above
;-----

[SourceDisksFiles]
mchpusb.sys=1
mchpusb64.sys=1

Línea 1, columna 1
```

Figura 5.9. Identificación de las dos tarjetas.

También fue necesario agregar un número identificador a cada tarjeta en el software de visual, puesto que si se quedaban con el mismo número solo se leería una y no sería posible manipular los dos robots.

6. Conclusiones y Trabajos futuros

6.1 Conclusiones

El LapBot junto con el Hibou son los primeros robots quirúrgicos construidos en la Universidad del Cauca que además formarán parte en un futuro de un entrenador quirúrgico que servirá para que estudiantes, docentes y profesionales de la ingeniería y la medicina se familiaricen con este tipo de sistemas.

Se puede concluir que la construcción de un robot debe ser cuidadosa y se deben tener en cuenta todos los parámetros existentes tanto matemáticos como físicos para lograr un funcionamiento óptimo de los sistemas.

Al finalizar este proyecto se pudo concluir que trabajar con la herramienta Solid Edge, para el diseño de las piezas del Lapbot fue muy conveniente ya que permitió el cálculo de los parámetros dinámicos del Lapbot, a partir de las masas y de los momentos de inercia respectivos a cada motor. Por otro lado daba la posibilidad de tener una perspectiva más cercana del robot en la vida real. Solid Edge tiene ciertas desventajas entre ellas que es un software poco intuitivo, lo que retrasó un poco el trabajo que se iba a desarrollar con él. Este aspecto se visualizó principalmente a la hora del ensamble de las piezas, labor que fue bastante compleja debido a los requerimientos de dicho software.

El micro controlador utilizado ofrece varias ventajas como por ejemplo su programación sencilla y la comunicación USB, sin embargo los puertos ofrecidos por este eran menos de los necesarios para el desarrollo del proyecto lo que generó mayor complejidad al momento de desarrollar el software.

Finalmente el ensamblaje final del robot y las pruebas realizadas desde el software mostraron un sistema, que aunque es susceptible de mejorar en muchos aspectos, se convierte en punto de partida para la construcción de nuevos y más elaborados prototipos.

6.2 Trabajos futuros

Como trabajos futuros se proponen los siguientes aspectos:

- Desarrollar el control por realimentación para el robot Lapbot, implementando un PID.
- Perfeccionar el material de construcción para obtener mejoras tanto en la parte estética como en el funcionamiento del robot, con la utilización de un material más liviano y pulido como por ejemplo acrílico.

- Elegir motores de alto torque pero de menor tamaño, para que así la estructura del Lapbot sea mucho más estética y funcional.
- Considerar algún tipo de mecanismo para prevenir la inercia, si la construcción se realiza con un material más liviano.
- Mejor diseño de las piezas para que el robot tenga una mejor apariencia y se asemeje mucho más a los robots quirúrgicos que existen en el mercado.
- Hacer más amigable la interfaz del software, que proporcione botones de ayuda, nombres describiendo que hace cada botón, letrero de bienvenida, opciones de registro de usuarios, creando una base de datos que los contenga; todo esto porque los usuarios finales son médicos o estudiantes de medicina.
- Trabajar con una tarjeta de adquisición de datos Arduino, que ofrece múltiples ventajas entre ellas que es una plataforma de hardware libre, cuyo entorno de desarrollo es sencillo de manipular y se adapta al desarrollo de proyectos multidisciplinarios.
- Utilizar FPGA como un medio alternativo y eficiente para la manipulación del robot, ya que ofrece múltiples ventajas en cuanto a ruido, problemas de interferencia, conexiones entre otros aspectos relacionados con el hardware.
- Desarrollar un entrenador quirúrgico tele operado para manejar el sistema de entrenamiento quirúrgico de los robots Lapbot e Hibou.

REFERENCIAS

1. P. Ricci, R. Lema, V. Solà, J. Pardo, E. Guiloff, "Desarrollo de la Cirugía Laparoscópica: pasado, presente y futuro. Desde Hipócrates hasta la introducción de la robótica en laparoscopia ginecológica", Revista Chilena de Obstetricia y Ginecología Vol. 73, pp. 63-75, 2008.
2. A. Carvajal, "Cirugía Robótica", Revista Cirujano General, Vol. 25, No. 4, pp. 314-320, 2003.
3. S. Salinas, "Modelado, Simulación en 3D y control de un Robot para cirugía Laparoscópica", Tesis de Maestría, Universidad del Cauca, Colombia, 2009.
4. A. Vivas, "Diseño y Control de robots industriales: Teoría y Práctica" Buenos Aires: Elaleph, 2010.
5. D. Stoianovici. "Robotic surgery". World Journal Urology. Vol. 18, pp. 289-295, 2000.
6. C. Arroyo, "Cirugía Robótica", Revista: Elementos Ciencia y Cultura, Vol. 12, pp. 13-17, 2005.
7. G. Tan, R. Goel, J. Kaouk, A. Tewari, "Avances tecnológicos en cirugía Laparoscópica", Clínicas Urológicas de Norteamérica Vol. 36, No. 2, pp. 237-250, 2009.
8. D. Murphy, B. Challacombe, T. Nedas, O. Elhage, K. Althoefer, L. Seneviratne, P. Dasgupta, "Equipamento y Tecnología en Robótica". Archivos Españoles de Urología, Vol. 60, No 4, pp. 349-355, 2007.
9. Intuitive Surgical. Da Vinci System. URL: <http://www.intuitivesurgical.com/specialties/>
10. M. Marohn and E. Hanly. "Twenty-first century surgery using twenty-first century technology:surgical robotics". CurrentSurgery. Vol. 61, pp. 466-473. 2004.
11. Surgery Enabled by da Vinci. URL:<http://www.davincisurgery.com/da-vinci-surgery/>.
12. Intuitive Surgical. Da Vinci S HD Surgical System. URL: http://www.intuitivesurgical.com/products/davinci_surgical_system/davinci_surgical_system_s/. Consultado: Julio de 2013.
13. Intuitive Surgical. Da Vinci Si HD Surgical System. URL:

http://www.intuitivesurgical.com/products/davinci_surgical_system/davinci_surgical_system_si/_Consultado: Julio de 2013.

14. Intuitive Surgical. Da Vinci Si HD Surgical System. Extended Features & Benefits. URL:
http://www.intuitivesurgical.com/products/davinci_surgical_system/davinci_surgical_system_si/features-benefits.html_Consultado: Julio de 2013.
15. J.A. Peña, M. Pascual, J.T. Salvador, A. Rosales, J. Palou y H. Villavicencio. "Evolución de la cirugía abierta versus laparoscópica/robótica: 10 años de cambios en Urología". Actas Urológicas Españolas, Vol. 34, No 3, pp. 223-231, 2009.
16. G. Dachs and W. Peine. "A Novel Surgical Robot Design: Minimizing the Operating Envelope within the Sterile Field". 28th IEEE EMBS Annual International Conference New York City, USA, Aug 30-Sept 3, 2006.
17. M. Acevedes. "Robótica y nuevos procedimientos en cirugía de obesidad". Cirujano General, Vol. 35, No. S1, pp. 52-56, 2013.
18. World Laparoscopy Hospital. Hands on Robotic Surgery Training. URL:
<http://www.laparoscopyhospital.com/roboticsurgerytraining.html>_Consultado: Julio de 2013.
19. Titan Medical Inc. URL: <http://www.titanmedicalinc.com/> Consultado: Julio de 2013.
20. La cirugía laparoscópica asistida por robot. URL:
http://www.marly.com.co/serv_cirugia_robotica.html_Consultado: Julio de 2013.
21. Colombia, a la vanguardia en cirugía robótica en la región. URL:
<http://www.vanguardia.com/actualidad/colombia/197245-colombia-a-la-vanguardia-en-cirugia-robotica-en-la-region>. Consultado: Julio de 2013.
22. Cirugía Laparoscópica Avanzada. C.L.A. URL:
<http://www.cirugialaparoscopicaavanzada.com/cirugia-laparoscopica.htm>__Consultado: Julio de 2013.
23. S. Schwartz y J. Hunter, "Principios de Cirugía", 7a ed, Vol. II. México: McGrawHill Interamericana, 1999.

24. A. Cescato. "Cirugía de Mínimo acceso: Posibilidades de la técnica laparoscópica en cirugía oncológica", Tesis de grado, Universidad Abierta Interamericana, Argentina, 2004.
25. E. Valle, "Experiencia en Cirugía Laparoscopia en HEODRA en el Periodo de Enero de 2003 a Febrero de 2004", Tesis de Especialización, Universidad Nacional Autónoma de Nicaragua, León, Nicaragua, 2004.
26. J. Yarmuch, H. Schulte, G. Caballero, C. Pinto, J. Silva y H. Silva, "Colecistectomía Laparoscópica, 1203 Casos con Colangiografía Ultraselectiva", Revista Española de Enfermería, Vol. 86, No. 2, pp. 587-581, 1994.
27. M. Meinero y G. Melotti, "Cirugía Laparoscópica". Buenos Aires: Editorial Médica Panamericana S.A., 1994.
28. R. Possik, E. Franco, D. Pires, D. Wohnrath and E. Ferreira, "Sensitivity, Specificity and Predictive Value of Laparoscopy for the Staging of Gastric Cancer and for the Detection of Liver Metastases", Cancer, Vol. 58, No. 1, pp. 1-6, 1986.
29. Surgical Alternatives to Hysterectomy in the Management of Leiomyomas. American College of Obstetricians and Gynecologists. Boletín No. 16, pp. 1-10, 2000.
30. J. Trussell, E. Guilbert and A. Hedley, "Sterilization Failure, Sterilization Reversal, and Pregnancy after Sterilization Reversal in Quebec", Obstetrics and Gynecology Journal, Vol. 101, pp. 677-684, 2003.
31. M. Menon, A. Tewari, B. Baize, B. Guillonneau and G. Vallancien, "Prospective Comparison of Radical Retropubic Prostatectomy and RobotAssisted Anatomic Prostatectomy: The Vattikuti Urology Institute Experience", Urology Journal, Vol. 60, pp. 864-868, 2002.
32. D. Crain, J. Roberts and C. Amling, "Practice Patterns in Vasectomy Reversal Surgery: Results of a Questionnaire Study Among Practicing Urologists", Journal Urology, Vol. 171, No. 1, pp. 311-315, 2004.
33. J. García, M. Arias, E. Valencia, "Diseño de prototipo de simulador para entrenamiento en cirugía laparoscópica", Revista Ingeniería Biomédica Vol. 5, No. 9, pp. 13-19, 2011.
34. J. Acosta, E. Muñoz, "Simulación virtual con tecnología Háptica: entrenamiento para la cirugía de rodilla", Repertorio de Medicina y Cirugía, Vol. 20, No. 4, pp. 245-250, 2011.

35. ProMIS: accelerating surgical performance through simulation. URL: <http://www.slideshare.net/Haptica/haptica-promis-sages-2009>. Consultado: Julio de 2013.
36. H. Delingette and N. Ayache, "Hepatic Surgery Simulation", Communications of the ACM, Vol. 48, pp. 31-36, No. 2, 2005.
37. O. Herrera, R. Espitia, G. Zayed, P. Figueroa, J. Torres, A. García, "Desarrollo de un sistema de entrenamiento médico para artroscopia de rodilla", Revista Colombiana de Ortopedia y Traumatología, Vol. 2, No. 3, pp. 172-177, 2008.
38. H. Schreuder, R. Wolswijk, R. Zweemer, M. Schijven, R. Verheijen, "Training and learning robotic surgery, time for a more structured approach: a systematic review", An International Journal of Obstetrics and Gynecology, Vol. 119, No. 2, pp. 137-149, 2012.
39. R. Valero, Y.H. Ko, S. Chauhan, O. Schatloff, A. Sivaraman, R.F. Coelho, F. Ortega, K.J. Palmer, R. Sanchez-Salas, H. Davila, X. Cathelineau, V.R. Patel, "Cirugía robótica: Historia e impacto en la enseñanza", Actas Urológicas Españolas, Vol. 35, No. 9, pp. 540-545, 2011.
40. dV-Trainer. Proven, cost-effective training for the da Vinci Surgical system. URL: <http://www.mimicsimulation.com/products/dv-trainer/>. Consultado: Julio de 2013.
41. Simulated Surgical Systems, LLC. Simulation for Robotic Surgery. URL: <http://www.simulatedsurgicals.com/what-is-ross.htm>. Consultado: Julio de 2013.
42. SimSurgery. URL: <http://www.simsurgery.com/>. Consultado: Julio de 2013.
43. C. Lallas, J. Davis, "Robotic surgery training with commercially available simulation systems in 2011: a current review and practice pattern survey from the society of urologic robotic surgeons", Journal of Endourology, Vol. 26, No. 3, pp. 283-293. 2012.
44. C. Méndez, V. Torres, "Diseño y Simulación en 3D de un Robot Porta Endoscopio para Cirugía Laparoscópica", Tesis de pregrado, Universidad del Cauca, Colombia, 2010.
45. C. Tosse, E. Lasso, "Dispositivo de Mando Auxiliar para el robot porta endoscopio Hibou", Tesis de pregrado, Universidad del Cauca, Colombia, 2012.
46. C. Muñoz, F. Ordoñez, "Software Manipulador del robot porta endoscopio Hibou para cirugía laparoscópica", Tesis de pregrado, Universidad del Cauca, Colombia, 2012.

47. Linak. Actuador lineal LA31 Homeline. URL: <http://www.linak.es/Productos/Linear-Actuators.aspx?product=LA31+HOMELINE>. Consultado: Septiembre de 2012.
48. M. Hadi, H. Zakaria, A. Almulhim, A. Alghamdi, H. Mowafi and A.Hussein, "Gas or Gas-Less Laparoscopic Cholecystectomy?", Kuwait Medical Journal, Vol. 34, pp. 292-295, 2002.
49. Motor Maxon EC power max 30. URL: <http://www.treffer.com.br/produtos/maxon/motores/pdf/188.pdf>. Consultado: Septiembre de 2012
50. Intuitive Surgical. "In Vivo Accessories for Minimally Invasive Robotic Surgery", U. S. Patent 7125403B2, 2006.
51. A. J. Madhani, "Design of Teleoperated Surgical Instruments for Minimally Invasive Surgery", PhD. thesis, Massachusetts Institute of Technology (MIT), United States, 1998.
52. Descriptors, Interfaces, Endpoints and Pipes. URL: <http://www.on-time.com/rtos-32-docs/rtusb-32/programming-manual/the-universal-serial-bus/descriptors-interfaces-endpoints-and-pipes.htm> Consultado: Mayo de 2013

ANEXOS

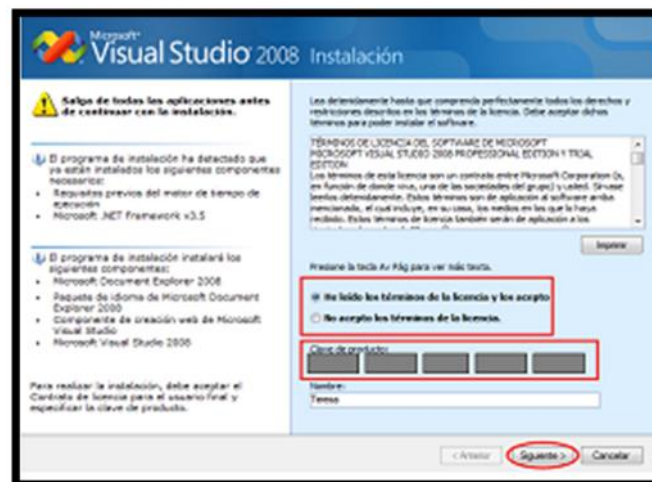
Anexo A. Instalación de las herramientas software utilizadas.

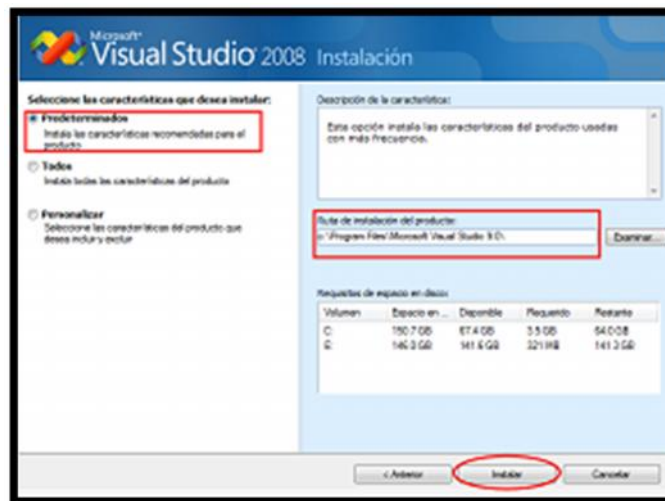
- **Instalación de Visual Studio 2008**

En las siguientes imágenes se mostrarán los pasos a seguir para instalar esta herramienta. El tipo de instalación será estándar.



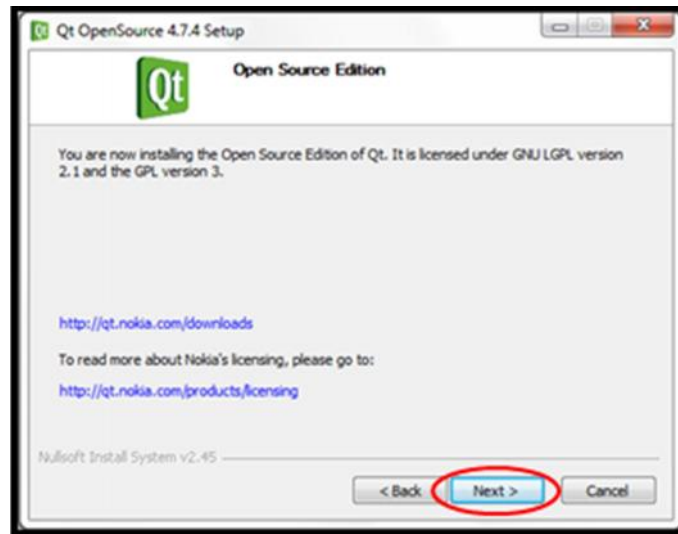
El instalador de tiene tres opciones de las cuales se escoge la primera “Instalar Visual Studio 2008” y a continuación se aceptan los términos y condiciones del programa. Después en las características a instalar se selecciona “Predeterminados”. Una vez realizado esto empieza la instalación del programa que durará varios minutos.

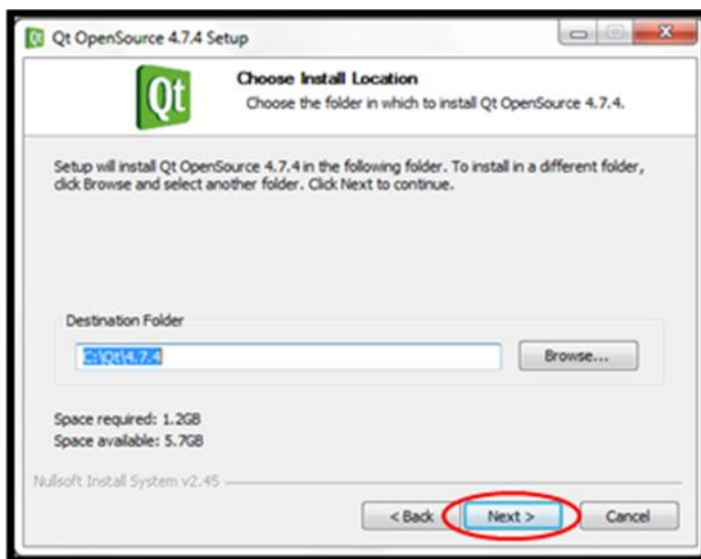
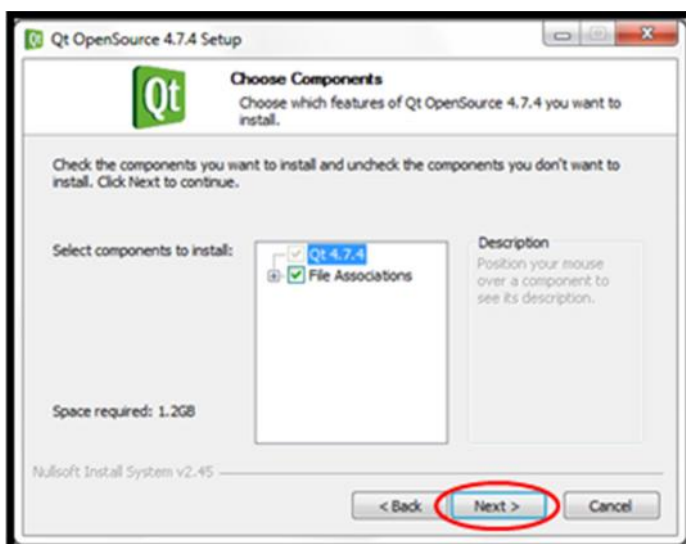
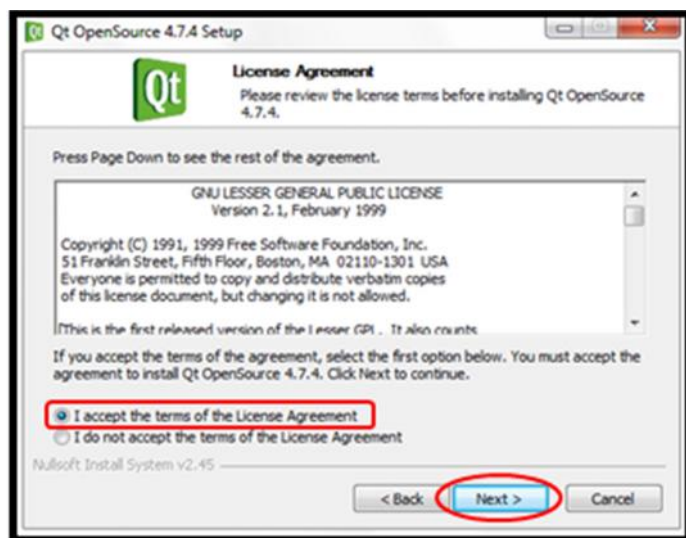


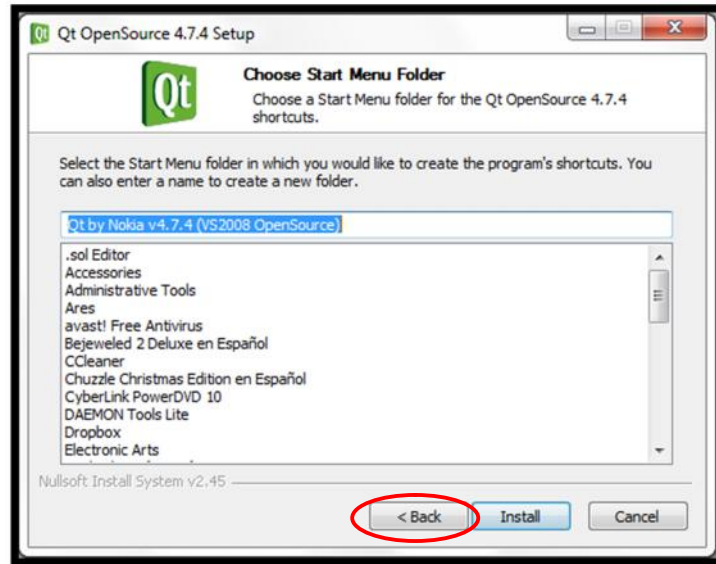


- **Instalación de Qt.**

Para el desarrollo de la interfaz se utilizó la versión “qt-win-opensource-4.7.4-vs2008” para generar la ventana secundaria donde se visualizan los dos robots. Este se descarga de la página <http://qt.nokia.com/downloads> y se instala con las opciones por defecto como se ve en las imágenes



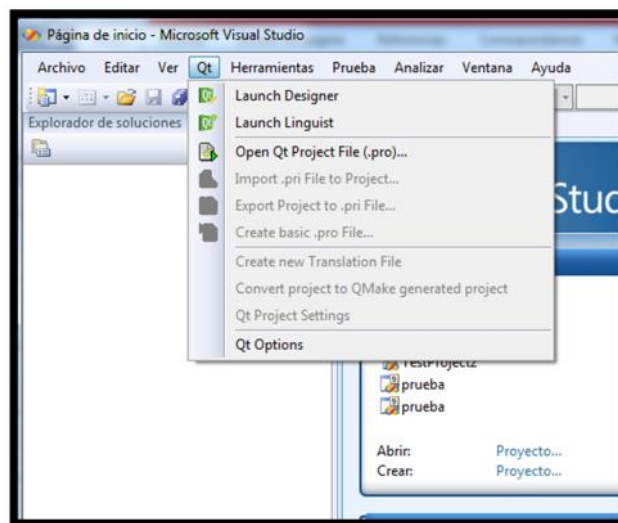




- **Integración de Qt con Visual Studio 2008.**

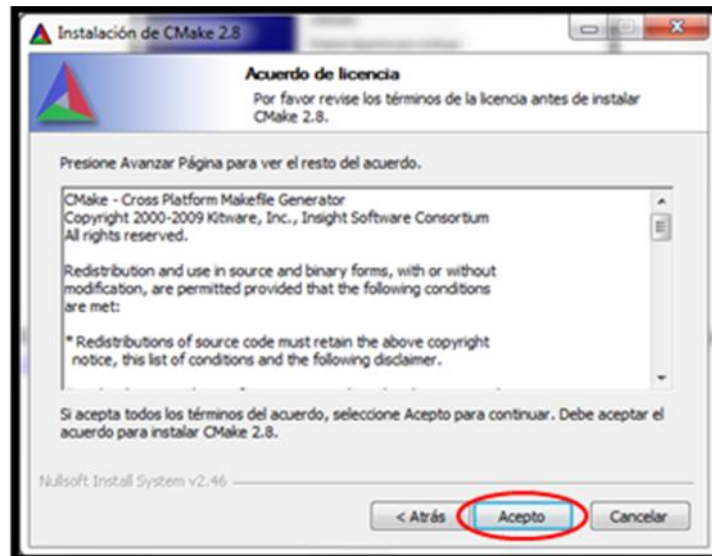
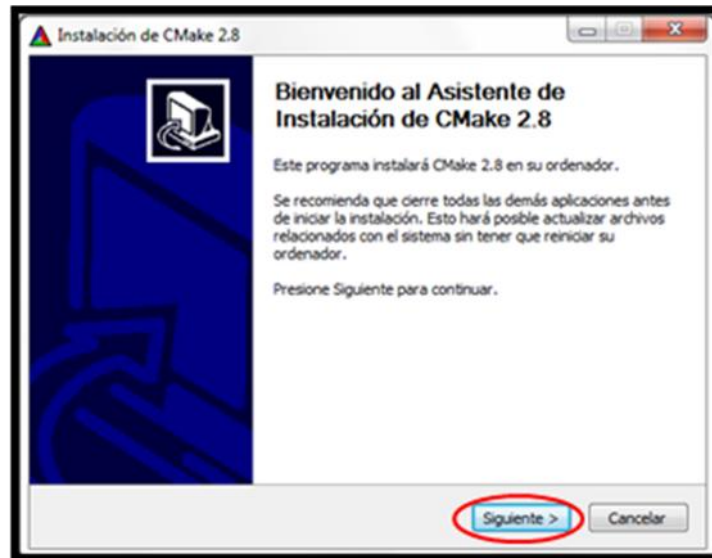
Para la integración de estas dos herramientas se hace uso del fichero “qt-vs-addin-1.1.9”, el cual se puede descargar a través de la página <http://qt.nokia.com/downloads/visual-studio-add-in>

Para comprobar la integración de estas dos herramientas se abre Visual Studio 2008 para verificar que aparece un nuevo menú llamado Qt.

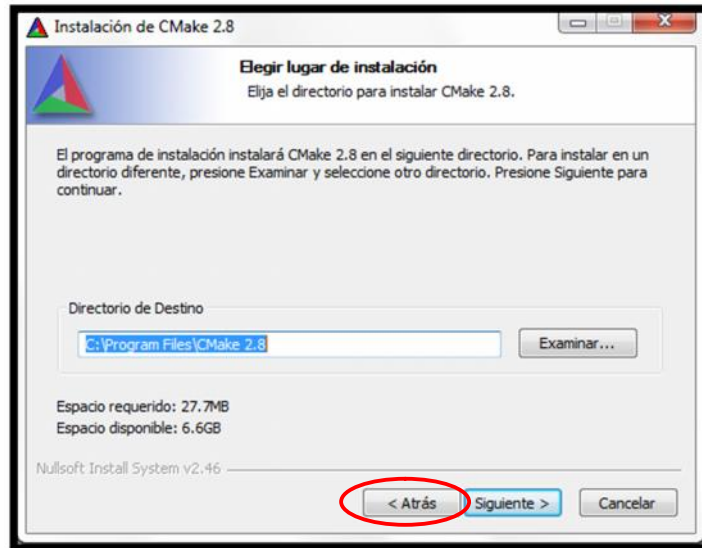
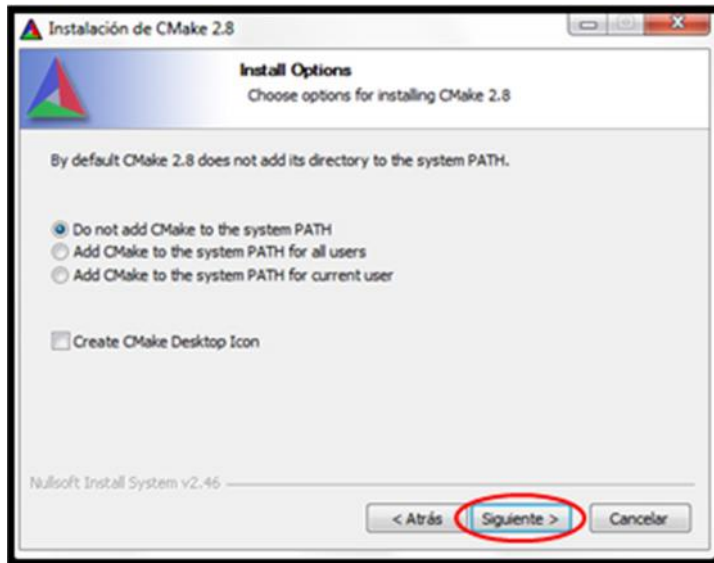


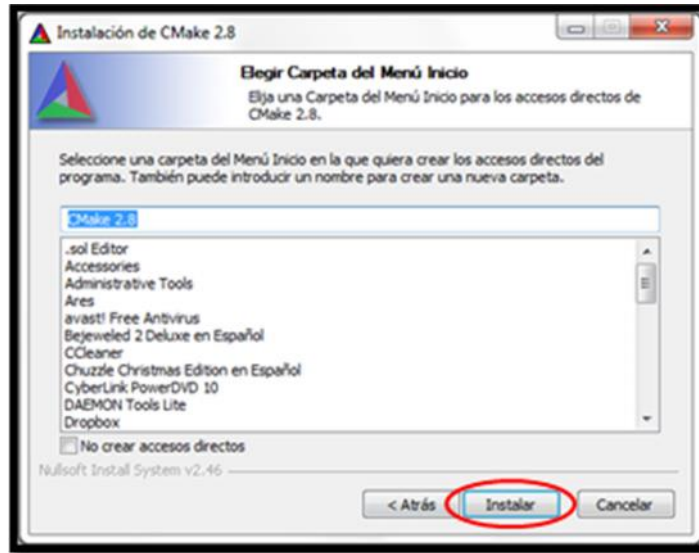
- **Instalación de Cmake.**

CMake es un compilador libre que se puede descargar de la página web <http://www.cmake.org/cmake/resources/software.html>. Se instalará la versión CMake 2.8 con las opciones predeterminadas como se muestra en las figuras.



Se debe escoger la opción “do not add CMake to the System PATH”, y se escoge la carpeta donde será instalado el programa.



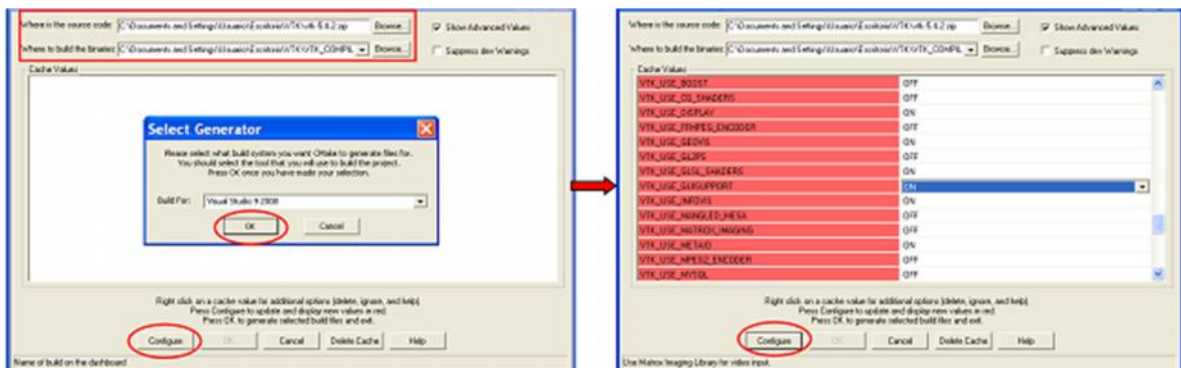


- **Instalación de VTK.**

VTK es una librería libre por lo tanto se puede descargar de su página web <http://www.vtk.org/VTK/resources/software.html>. Se descargará un archivo .zip el cual debe ser descomprimido.

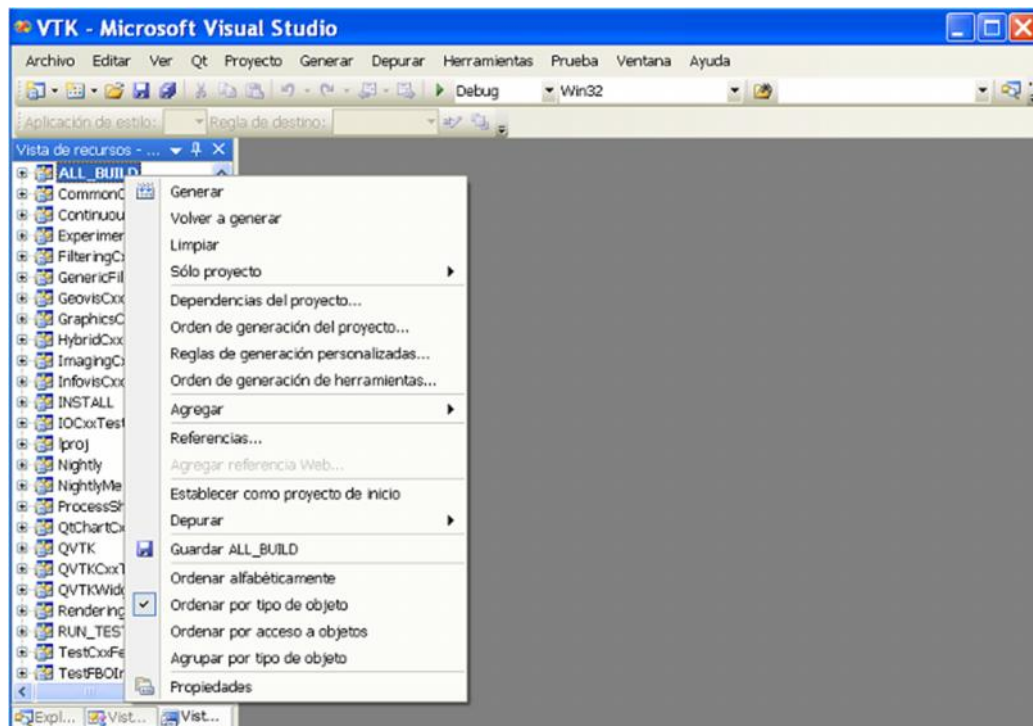
Se ejecuta la aplicación CMake para compilar la librería VTK, seguidamente en la ventana de Cmake donde indica *“Where is the source code”* se pone el directorio donde fue descomprimida la carpeta de VTK. Luego en *“Where to build the binaries”* se ubica la dirección de una nueva carpeta creada por el usuario donde quedará la librería VTK compilada que creará Cmake.

Después de esto se dará click en *“configure”* y saldrá un cuadro donde se pregunta por la plataforma donde se realizará la precompilación, ahí se escoge **Visual Studio 9**. Después de esto se dará click una vez más en *“configure”* y saldrán unas variables en rojo las cuales se pueden cambiar de *“on”* a *“off”*. En esta ventana se cambiará la variable **“VTK_USE_GUI SUPPORT”** de *“on”* a *“off”*.



Se da “configure” una vez más y después se procede a cambiar de “off” a “on” las variables “VTK_USE_QVTK”, “DESIRED_QT_VERSION” y “VTK_USE_QVTK_QTOPENGL”, cada vez que se activa una variable se debe dar click en “configure”, es decir que se debe hacer esto un total de cuatro veces, una por cada variable activada. Una vez finalizado el proceso las variables aparecerán en color gris y ya se podrá dar click en “ok”.

En el directorio que se introdujo cuando se preguntó “Where to build the binaries” aparece un proyecto llamado “Vtk.sln” que se ejecutará en Visual Studio 2008. Una vez abierto se situará el cursor donde dice “ALL_BUILD”, se dará click derecho y se escogerá la opción “Generar”. Este proceso tardará más de una hora. Las librerías se crearán en la carpeta donde está “Vtk.sln”. Una vez hecho esto las librerías se podrán incluir en cualquier proyecto.



- **Instalación de SDL.**

SDL es una librería de uso libre la cual se puede descargar de la página <http://www.libsdl.org/download-1.2.php>, se busca “Development Libraries” y se descarga el archivo “SDL-devel-1.2.15-VC.zip (Visual C++)” como se muestra en la figura. Una vez descargada esta carpeta se descomprime en el disco C.

SDL version 1.2.15 (historic)

Source Code:

What's New
 SDL-1.2.15.tar.gz - GPG signed
 SDL-1.2.15-1.src.rpm - GPG signed
 SDL-1.2.15.zip - GPG signed

Runtime Libraries:

Linux:
 SDL-1.2.15-1.i386.rpm
 SDL-1.2.15-1.x86_64.rpm

Win32:
 SDL-1.2.15-win32.zip
 SDL-1.2.15-win32-x64.zip (64-bit Windows)

Mac OS X:
 SDL-1.2.15.dmg (Intel 10.5+)
 SDL-1.2.15-OSX10.4.dmg (Intel/PPC 10.4)

Development Libraries:

Linux:
 SDL-devel-1.2.15-1.i386.rpm
 SDL-devel-1.2.15-1.x86_64.rpm

Win32:
SDL-devel-1.2.15-VC.zip (Visual C++)
 SDL-devel-1.2.15-mingw32.tar.gz (Mingw32)

GPG Signature:

The source code to this release has been signed by Sam Lantinga.
 You can get the public key from any keyservers with the key id 0xA77638E6, or directly from Sam's home page: sliouken-pubkey.asc
 The public key fingerprint should be:

pub 1024017FF888 2010-10-02 Sam Lantinga <slantinga@lunarg.com>
 New fingerprint = 10A8 8008 A038 807F 7763 8038 5308 807F A776 38E6

For more information about public key signatures, see <http://www.gnupg.org/>

La ruta a seguir en el equipo será: **Panel de control >> Sistema y Seguridad >> Sistema**. Se abre la opción **Configuración avanzada del sistema**, la cual desplegará una ventana llamada **Propiedades del sistema**. Aquí se abre la opción **Variables de entorno** que desplegará una ventana de variables de entorno. En esta ventana se da click en el botón **Nueva**, en esta nueva ventana se agrega la nueva variable de entorno, como nombre tendrá **SDLDIR** y en el valor de la variable se escribe **C:\SDL-devel-1.2.15-VC\SDL-1.2.15**.

El anterior procedimiento se muestra en las siguientes imágenes:



Centro de actividades
 Revisar el estado del equipo y resolver los problemas | Cambiar configuración de Control de cuentas de usuario | Solucionar problemas habituales del equipo
 Restaurar el equipo a un punto anterior en el tiempo

Firewall de Windows
 Comprobar estado del firewall | Permitir un programa a través de Firewall de Windows

Sistema
 Ver la cantidad de memoria RAM y la velocidad del procesador | Consultar la Evaluación de la experiencia en Windows | Permitir acceso remoto
 Mostrar el nombre de este equipo | Administrador de dispositivos

Windows Update
 Activar o desactivar la actualización automática | Buscar actualizaciones | Ver actualizaciones instaladas

Opciones de energía
 Cambiar la configuración de batería | Requerir una contraseña cuando el equipo se reactiva | Cambiar las acciones de los botones de encendido | Cambiar la frecuencia con la que el equipo entra en estado de suspensión

Copias de seguridad y restauración
 Hacer una copia de seguridad del equipo | Restaurar archivos desde copia de seguridad

Cifrado de unidad BitLocker
 Proteger el equipo cifrando los datos en el disco | Administrar BitLocker

Herramientas administrativas
 Liberar espacio en disco | Desfragmentar el disco duro | Crear y formatear particiones del disco duro | Ver registros de eventos | Programar tareas

Flash Player

Ventana principal del Panel de control

- Administrador de dispositivos
- Configuración de Acceso remoto
- Protección del sistema
- Configuración avanzada del sistema**

Ver información básica acerca del equipo

Edición de Windows
 Windows 7 Ultimate
 Copyright © 2009 Microsoft Corporation. Reservados todos los derechos.

Propiedades del sistema

Nombre de equipo | Hardware

Opciones avanzadas | Protección del sistema | Acceso remoto

Para realizar la mayoría de estos cambios, inicie sesión como administrador.

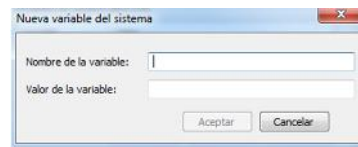
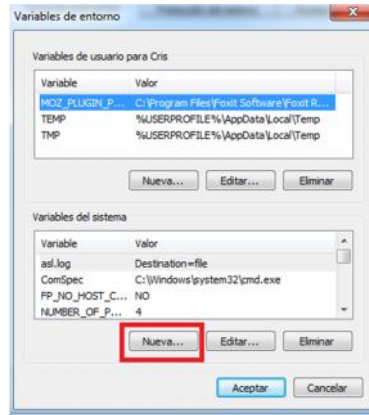
Rendimiento
 Efectos visuales, programación del procesador, uso de memoria y memoria virtual
 Configuración...

Perfiles de usuario
 Configuración del escritorio correspondiente al inicio de sesión
 Configuración...

Inicio y recuperación
 Inicio del sistema, errores del sistema e información de depuración
 Configuración...

Variables de entorno...

Aceptar | Cancelar | Aplicar



Anexo B. Manual de usuario del sistema.

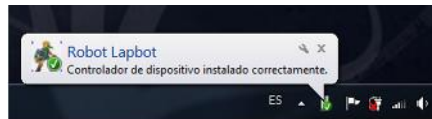
Introducción

Este manual describe los pasos que se deben seguir para poner en funcionamiento el sistema, el manual enumera tanto los pasos que se deben seguir en el computador como los que se deben aplicar al hardware.

Paso 1

Conectar la tarjeta de adquisición de datos mediante el cable de conexión USB. Previamente se debe haber instalado el driver del usb. Para esto se copia la carpeta que viene con los documentos del proyecto y se pega en la dirección deseada, así cuando se conecte el usb por primera vez, este verificará que exista el driver y quedará instalado en el computador.

Una vez conectada la tarjeta y con el driver instalado, debe aparecer un mensaje en la esquina inferior derecha mostrando que el dispositivo se ha reconocido.



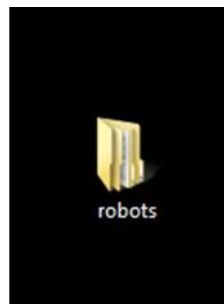
Además se debe conectar el joystick a través del cual se manipulará el robot.

Paso 2

Realizar el encendido de las fuentes que permitirán el funcionamiento del robot, de los potenciómetros y de la tarjeta de adquisición. Esto se realizará a través de un botón de encendido y apagado ubicado en la parte inferior de la mesa de trabajo.

Paso 3

Buscar la carpeta donde se encuentra el archivo ejecutable de la interfaz. Esta carpeta se encuentra en el escritorio con el nombre **robots**.



Aquí se encuentran tres carpetas, seleccionamos la carpeta con el nombre **robotic**.

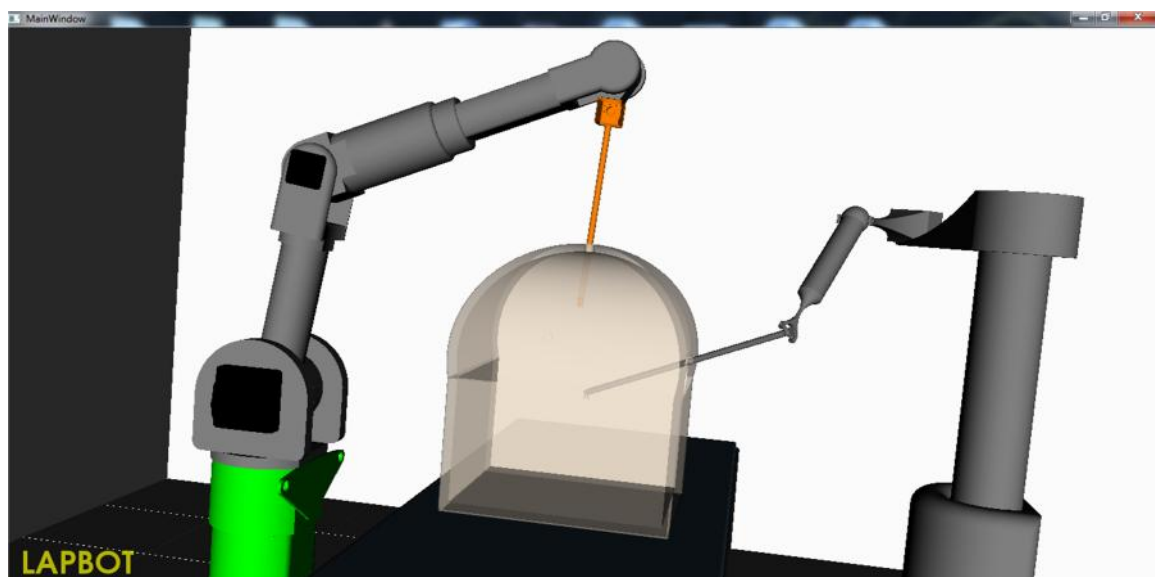
Nombre	Fecha de modifica...	Tipo
piezas	09/09/2013 03:18 ...	Carpeta de archivos
robotic	09/09/2013 03:17 ...	Carpeta de archivos
robots	09/09/2013 03:17 ...	Carpeta de archivos

Una vez aquí se busca la carpeta **Debug** y dentro de esta se encuentra el archivo ejecutable que contiene una extensión **.exe**. El nombre del ejecutable es **APLICACION**.

Nombre	Fecha de modifica...	Tipo	Tamaño
APLICACION	05/09/2013 05:41 ...	Aplicación	12.690 KB
aplicacion	05/09/2013 05:41 ...	VC++ Minimum R...	2.483 KB
APLICACION	05/09/2013 05:41 ...	Incremental Linke...	17.301 KB
aplicacion	05/09/2013 05:41 ...	Program Debug D...	39.004 KB
usb2550.dll	13/12/2005 01:06 a...	Extensión de la apl...	32 KB

A este archivo se le da click derecho y se escoge la opción “ejecutar como administrador” y se responde *sí* a la pregunta del sistema.

Una vez realizado esto aparecerá en pantalla una interfaz mostrando el sistema con los dos robots: LapBot e Hibou.



Con esto, el sistema está listo para funcionar y el robot real seguirá los movimientos del robot virtual.

Este manual se puede seguir para verificar el movimiento de los dos robots al tiempo. No hay modificación en el seguimiento de los pasos pero cabe recordar que se debe verificar que el driver de la tarjeta sea el indicado y este correctamente instalado.