

**DISEÑO Y VALIDACIÓN DE UN EXOESQUELETO DE PIERNAS DE TIPO MAESTRO-ESCLAVO  
PARA FACILITAR LA REHABILITACIÓN DE PERSONAS CON DISCAPACIDAD PARCIAL DE SU  
LOCOMOCIÓN EN EL PLANO SAGITAL**

**JUAN MANUEL GROSSO PÉREZ**

**Proyecto de grado para obtener el título de Ingeniero Mecatrónico**

**Director:  
Mg. Ing. Diego Alexander Tibaduiza**

**UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA  
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS  
INGENIERÍA MECATRÓNICA  
BUCARAMANGA  
2008**

Nota de aceptación:

---

---

---

---

---

Firma del presidente del jurado

---

Firma del jurado

---

Firma del jurado

Bucaramanga, 12 de mayo de 2008

## Tabla de contenido

INTRODUCCIÓN .....	9
<b>1. TECNOLOGÍAS PARA ASISTIR LA MOVILIDAD PERSONAL .....</b>	<b>11</b>
1.1 ESTIMULACIÓN ELÉCTRICA FUNCIONAL (EEF) .....	11
1.2 ORTESIS Y EXOESQUELETOS .....	12
1.2.1 Ortesis pasivas .....	13
1.2.2 Ortesis activas.....	14
1.2.3 Trabajos relacionados .....	16
<b>2. ASPECTOS GENERALES DE LA MARCHA HUMANA.....</b>	<b>19</b>
2.1 CICLO DE MARCHA.....	19
2.1.1 Fase de apoyo.....	20
2.1.2 Fase de balanceo .....	20
2.2 CINEMÁTICA .....	21
2.2.1 Cinemática angular de la cadera .....	23
2.2.2 Cinemática angular de la rodilla .....	23
2.2.3 Cinemática angular del tobillo .....	24
2.3 TÉCNICAS PARA CAPTURA DE MOVIMIENTO CORPORAL.....	25
2.3.1 Acelerometría.....	25
2.3.2 Análisis por video y fotografía .....	26
2.3.3 Electrogoniometría .....	27
<b>3. SISTEMA DE MONITOREO Y ASISTENCIA DE LA MARCHA (LEGXOS) .....</b>	<b>29</b>
3.1 REQUERIMIENTOS Y ESPECIFICACIONES.....	30
<b>4. MODELAMIENTO DEL SISTEMA .....</b>	<b>34</b>
4.1 DATOS ANTROPOMÉTRICOS .....	35
4.1.1 Longitud de los segmentos .....	35
4.1.2 Masa y centro de masas .....	36
4.2 DINÁMICA.....	39
4.2.1 Formulación Lagrangiana .....	40
4.2.2 Formulación Newton-Euler .....	46
4.2.3 Modelamiento gráfico con Matlab/Simulink y SimMechanics .....	49
<b>5. DISEÑO ELECTRÓNICO .....</b>	<b>58</b>
5.1 SENSORES .....	59
5.1.1 Caracterización del sensor utilizado .....	60
5.2 UNIDAD DE PROCESAMIENTO LOCAL.....	62
5.2.1 Comunicación Serial RS-232.....	63
5.2.2 Conversión Análogo-Digital.....	67
5.2.3 Control de servomotores por modulación del ancho de pulso (PWM) .....	68
5.3 FUENTES DE ENERGÍA .....	70

<b>6. DISEÑO MECÁNICO.....</b>	<b>72</b>
6.1 DISEÑO CONCEPTUAL .....	72
6.1.1 Diseño basado en actuador lineal: .....	73
6.1.2 Diseño basado en servomotores .....	74
6.1.3 Selección del actuador .....	74
6.1.4 Diseño final del exoesqueleto .....	75
6.2 MODELO DE VALIDACIÓN .....	79
6.2.1 Modelado 3D .....	80
6.2.2 Dimensionamiento y selección de los servomotores .....	81
<b>7. DESARROLLO DEL SOFTWARE.....</b>	<b>83</b>
7.1 INTERFAZ DEL SISTEMA LEGXOS.....	83
7.1.1 Programación de la unidad local basada en PIC16F873A .....	84
7.1.2 Interfaz Gráfica de Usuario .....	85
7.1.3 Simulaciones en 2D y 3D .....	91
<b>8. PRUEBAS Y RESULTADOS.....</b>	<b>94</b>
<b>9. OBSERVACIONES Y CONCLUSIONES.....</b>	<b>101</b>
<b>10. Bibliografía .....</b>	<b>103</b>
<b>ANEXOS .....</b>	<b>104</b>

## LISTA DE FIGURAS

Figura 1. Estimulación Eléctrica Funcional aplicada al control del ángulo de la rodilla .....	12
Figura 2. Ortesis de marcha recíprocante (RGO).....	13
Figura 3. Ejemplos de actuadores en ortesis activas. ....	15
Figura 4. Reseña de exoesqueletos .....	17
Figura 5. Ciclo de la marcha humana .....	19
Figura 6. Eventos de la fase de apoyo.....	20
Figura 7. Eventos de la fase de balanceo .....	21
Figura 8. Planos de referencia para la cinemática humana .....	22
Figura 9. Ángulos relativos entre los segmentos del miembro inferior .....	22
Figura 10. Flexión y extensión de cadera durante la marcha. ....	23
Figura 11. Flexión y extensión de rodilla durante la marcha. ....	24
Figura 12. Flexión dorsal y flexión plantar del tobillo durante la marcha. ....	25
Figura 13. Ubicación esquemática de acelerómetros para medir rotación articular.....	26
Figura 14. Sistema óptico para captura de movimiento.....	27
Figura 15. Análisis de marcha usando un electrogoniómetro .....	28
Figura 16. Esquema general del proceso de diseño mecatrónico adoptado.....	31
Figura 17. Diagrama de bloques del sistema de monitoreo y asistencia de la marcha. ....	32
Figura 18. Modelo de una pierna con dos grados de libertad. ....	34
Figura 19. Longitud de los segmentos de la pierna en función de la altura total. ....	36
Figura 20. Librerías de SimMechanics dentro del entorno Simulink de Matlab .....	50
Figura 21. Modelado del sistema muslo-pantorrilla en SimMechanics.....	52
Figura 22. Bloques de entrada y salida .....	52
Figura 23. Propiedades del bloque <i>Ground</i> .....	53
Figura 24. Propiedades del bloque <i>Body</i> .....	53
Figura 25. Propiedades del bloque <i>Revolute</i> .....	54
Figura 26. Propiedades del bloque <i>Joint Initial Condition</i> y <i>Machine Environment</i> .....	55
Figura 27. Propiedades de los bloques <i>Sensor</i> y <i>Actuator</i> .....	56
Figura 28. Dinámica para el sistema muslo-pantorrilla en fase de balanceo. ....	57
Figura 29. Etapas electrónicas del sistema LEGXOS .....	58
Figura 30. Potenciómetro rotacional .....	59
Figura 31. Respuesta experimental del sensor de posición angular empleado.....	62
Figura 32. Diagrama de comunicación serial entre pc y microcontrolador .....	64
Figura 33. Módulos de Radiofrecuencia empleados .....	66
Figura 34. Diagrama de bloques de un servomotor dc. ....	68
Figura 35. Ciclos útiles de los servomotores Hitec empleados. ....	69
Figura 36. Diseño conceptual de las propuestas iniciales.....	72
Figura 37. Funcionamiento de la propuesta con actuador lineal.....	73

Figura 38. Diseño final de cada propuesta. ....	74
Figura 39. Imagen 3D del diseño desarrollado. ....	76
Figura 40. Propiedades físicas del diseño. ....	77
Figura 41. Cadera. ....	78
Figura 42. Estructura activa. ....	78
Figura 43. Estructura pasiva. ....	79
Figura 44. Diseño 3D de la plataforma bípeda experimental. ....	80
Figura 45. Diagrama de cuerpo libre para el dimensionamiento de motores ....	81
Figura 46. Diseño 3D de los servomotores. ....	82
Figura 47. Diagrama de flujo del microcontrolador para adquisición de datos. ....	84
Figura 48. Interfaz gráfica en MATLAB para el sistema LEGXOS. ....	85
Figura 49. Paleta principal de la interfaz. ....	86
Figura 50. Menú principal de la interfaz. ....	86
Figura 51. Menú desplegable para el registro de nuevos pacientes. ....	87
Figura 52. Lista de pacientes registrados. ....	88
Figura 53. Mensajes de alerta en la opción eliminar. ....	88
Figura 54. Mensaje de alerta al no establecer comunicación. ....	89
Figura 55. Menú Visualizaciones y modos de operación. ....	89
Figura 56. Opción Abrir archivo. ....	90
Figura 57. Opción Guardar archivo. ....	91
Figura 58. Fase de balanceo en entorno de realidad virtual. ....	92
Figura 59. Fase de balanceo en entorno de SimMechanics. ....	92
Figura 60. Modelo de cuerpos rígidos en SimMechanics para la marcha. ....	93
Figura 61. Comparación de respuestas del convertidor A/D. ....	96
Figura 62. Seguimiento de un ejercicio de flexión y extensión de la cadera ....	97
Figura 63. Seguimiento de un ejercicio de flexión y extensión de la rodilla. ....	98
Figura 64. Seguimiento de la plataforma de un ejercicio de elevación de la pierna. ....	98
Figura 65. Seguimiento de la plataforma en un ciclo de marcha. ....	99
Figura 66. Resultados del modelo experimental. ....	100

## LISTA DE TABLAS

Tabla 1. Datos antropométricos para miembro inferior.....	37
Tabla 2. Datos experimentales de la respuesta del sensor.....	61
Tabla 3. Criterios de selección de la solución. (A) Servomotores, (B) Actuador lineal.....	75

## LISTA DE ANEXOS

ANEXO A. Tablas de D. Winter para la cinemática de un ciclo de marcha. ....	105
ANEXO B. Esquemáticos .....	108
ANEXO C. Especificaciones de los Módulos RF.....	109
ANEXO D. Especificaciones y operación de la tarjeta controladora de servos .....	110
ANEXO E. Planos de los diseños mecánicos .....	123
ANEXO F. Especificaciones de los servomotores .....	137
ANEXO G. Códigos fuente de los programas.....	142



## INTRODUCCIÓN

Desde tiempos remotos grandes artistas, científicos, inventores y otro gran cúmulo de profesionales de diversas áreas han mostrado gran fascinación por el estudio de la anatomía y de la locomoción animal, en especial de la marcha bípeda, que se establece como una característica esencial de la condición del ser humano, y ha sido causa fundamental en el desarrollo de conceptos, técnicas y tecnologías que buscan describir, analizar, y emular la biomecánica de sistemas biológicos. Tal es el interés en aprovechar las capacidades locomotoras del hombre, que hoy en día integrar a los humanos con las máquinas en un sistema, ofrece múltiples oportunidades para crear nuevas aplicaciones mecatrónicas.

Los humanos presentan complejos y especializados algoritmos naturales de control, que les brindan la capacidad de realizar tareas complicadas en un amplio rango de condiciones y con rápidos tiempos de respuesta. En contraste, los robots pueden desarrollar tareas que requieren grandes fuerzas o torques, dependiendo de la naturaleza de su estructura y de la potencia de sus actuadores; sin embargo, sus algoritmos artificiales de control que gobiernan la dinámica del mecanismo, pierden en gran medida la flexibilidad y calidad de actuación que tienen los humanos de realizar tareas en ambientes difusos con condiciones imprecisas. Es evidente entonces, que combinando estas dos entidades, el humano y el robot, en un solo sistema integrado, se pueden alcanzar soluciones interesantes que se beneficiarían de las ventajas que aporta cada subsistema. Así, la potencia mecánica de las máquinas integrada con el sistema de control inherente al humano, llevaría a realizar tareas que necesitan aplicar grandes fuerzas de una manera eficiente, lo cual es el principio básico del diseño de sistemas exoesqueléticos.

Ahora bien, si se tiene presente que es a través de la locomoción natural que el ser humano goza de autonomía física y ejecuta la mayor parte de sus actividades cotidianas, se puede identificar a su vez la problemática social que se genera al ser afectada dicha capacidad en los individuos, ya sea por factores tales como enfermedades, vejez, accidentes o guerras; lo cual trae consigo el detrimento del autoestima de la persona, la limitación de su desempeño dentro de la sociedad y en algunos casos la dependencia total de un tercero. Para ilustrar esta situación en valores cuantificables, los indicadores nacionales registrados oficialmente por el DANE<sup>1</sup>, señalan que para el año 2007 existían

---

<sup>1</sup> Departamento Administrativo Nacional de Estadísticas (DANE). *Índice de cuadros de la población con registro para la localización y caracterización de las personas con discapacidad (2007)*.

en Colombia alrededor de 300 mil personas con limitaciones físicas, parciales o permanentes, relacionadas con la motricidad de las piernas.

De esta manera, ciertas áreas como la medicina, las ciencias computacionales y la mecatrónica se han visto en la necesidad de converger sus conocimientos en torno al desarrollo de aplicaciones relativas a la acción de caminar, logrando complementarse y obtener grandes resultados como el monitoreo y diagnóstico de patologías de la marcha, estrategias eficientes de rehabilitación, creación de prótesis y ortesis, simulaciones y animaciones virtuales de la marcha, diseño y construcción de robots bípedos, y más recientemente, el desarrollo de tecnologías para locomoción y rehabilitación asistida, el cual es el marco de aplicación en el cual se centra el presente proyecto.

De la población con limitaciones físicas, un porcentaje significativo sufre específicamente de atrofia o distrofia en una sola de sus extremidades inferiores mientras la otra permanece sana; en consecuencia, surge un problema bien delimitado y posible de abordar, enmarcado en el requerimiento de un sistema que permita compensar la locomoción de la pierna afectada, y recuperar así, el valor funcional del afectado dentro de la sociedad.

Es por ello que esta investigación se orienta en contribuir con la aplicación de ingeniería en el desarrollo de dispositivos para sistemas de asistencia en la rehabilitación de la marcha humana. Específicamente el trabajo presentado aprovecha la simetría existente en la marcha humana para diseñar y validar un sistema mecatrónico de exoesqueleto de tipo maestro-esclavo, que permite asistir la marcha de una persona con discapacidad en una de sus piernas, tomando como referencia el movimiento de la pierna sana. Cabe recalcar que este proyecto se limita sólo al movimiento en el plano sagital.

Para la consecución de este proyecto se empleó la metodología del diseño mecatrónico basado en un sistema dinámico de investigación, modelado y simulación. El trabajo se ha organizado de tal manera que en sus primeros capítulos se presente al lector una breve ambientación sobre el marco teórico de referencia conformado por un estado del arte de los avances en sistemas exoesqueléticos, así como la caracterización de la marcha humana y una serie de conceptos fundamentales de robótica y herramientas matemáticas. Posterior a ello, se describe el desarrollo del sistema mecatrónico concebido, integrado por sus partes electrónicas, mecánicas y computacionales. Finalmente se exponen los resultados y alcances obtenidos.

# **1. TECNOLOGÍAS PARA ASISTIR LA MOVILIDAD PERSONAL**

Las tecnologías de asistencia para locomoción son aquellos dispositivos enfocados en brindar a las personas con discapacidad motriz, alternativas diferentes a las ya tradicionales sillas de ruedas, caminadores, muletas y bastones. Estos dispositivos de movilidad asistida son, como su nombre lo indica, para contribuir en el movimiento del usuario, y su éxito se mide a través de su habilidad para incrementar los niveles funcionales de la persona [ (WESLEY, 2005)]. Actualmente, existen dos áreas de ayudas para la movilidad personal fuertemente investigadas y en desarrollo, éstas son: la Estimulación Eléctrica Funcional (EEF) y las Ortesis activas y exoesqueletos [ (G. LOEB, 2000)]. De esta manera, el propósito de este primer capítulo es presentar de forma general los conceptos ligados a estas dos tecnologías, especialmente aquellos relacionados con las ortesis activas, con el fin de ambientar al lector en el principal objetivo de este trabajo.

## **1.1 ESTIMULACIÓN ELÉCTRICA FUNCIONAL (EEF)**

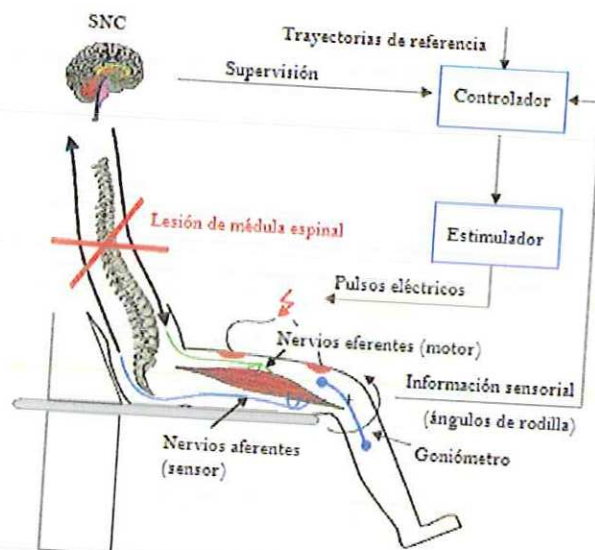
La estimulación eléctrica es una técnica que permite generar y controlar la contracción de un músculo como respuesta a una corriente eléctrica inducida de manera externa al sistema nervioso central, permitiendo así que se puedan excitar músculos cuyas comunicaciones nerviosas se hayan dañado parcial o totalmente. A partir de 1961, Liberson adaptó esta técnica para su uso como ortesis electrónica con fines funcionales [ (SOTELANO, 2003)], motivo por el cual recibió la denominación de estimulación eléctrica funcional (EEF) que se aplica cuando la corriente eléctrica logra contraer al músculo con la fuerza necesaria y en el momento adecuado para que éste pueda cumplir alguna función [ (MARSOLIS, 1983)], en el caso de interés, caminar. En la figura 1, se ilustra el principio de la EEF en un problema específico, como lo es el control del ángulo de la rodilla por estimulación del cuádriceps.

Desde sus inicios, los trabajos de investigación relativos a la EEF, han estado enfocados a identificar los patrones de los impulsos eléctricos que permitan, de manera eficiente, lograr una locomoción funcional. Entre los problemas que surgieron y que fijaron la atención de los investigadores, se pueden mencionar la falta de control en la biomecánica, la aparición rápida de fatiga muscular, la dificultad para identificar la estrategia de estimulación muscular y la manera de seleccionar a los candidatos para este tipo de terapia, y la confiabilidad de los sistemas [ (VILLANUEVA D., 2001)].

Por otro lado, los resultados obtenidos hasta el momento no son estables, en algunos pacientes son grandes los aportes y en otros casos no se logran beneficios significativos.

De esta manera, el uso de EEF para asistir la marcha aún no se ha extendido, sólo algunos centros de investigación, en países desarrollados, cuentan con hospitales donde es posible tener acceso a esta alternativa de rehabilitación, pero todavía como una fase experimental. En el ámbito mundial, existen muy pocas compañías que ofrecen comercialmente sistemas de EEF para la recuperación de la locomoción funcional.

Figura 1. Estimulación Eléctrica Funcional aplicada al control del ángulo de la rodilla



Fuente: Instituto Max Planck para dinámica de sistemas técnicos complejos. (Adaptado de *Controlled Functional Electrical Stimulation (FES) in the Rehabilitation of Spinal Cord Injured Persons and Stroke*).

## 1.2 ORTESIS Y EXOESQUELETOS

El otro enfoque que actualmente se desarrolla para minimizar las limitaciones motrices es el uso de ortesis, definidas por la Organización Internacional de Estándares como un aparato usado para modificar las características funcionales o estructurales del sistema neuronal-muscular-esquelético [ (BOWKER P, 1993)], o según el diccionario médico, Dorland's Illustrated Medical Dictionary, como un dispositivo o aparato utilizado para soportar, alinear, prevenir, corregir deformidades o mejorar el movimiento de alguna parte del cuerpo.

En otras palabras, se podría decir que una ortesis es la combinación e integración, entre las partes del cuerpo y una pieza de ingeniería, donde el resultado de esa integración es una unidad que obedece las leyes de la física y logra efectos o beneficios biomecánicos.

Los materiales más comúnmente usados en la fabricación del marco estructural de las ortesis son: acero inoxidable, aluminio, polímeros y velcro. En algunos casos se utiliza titanio y grafito. Sin embargo, algunas aleaciones de aluminio han demostrado tener una buena relación peso/resistencia, lo cual lo hace un material muy apropiado para esta aplicación, a un precio accesible. Por su parte, los polímeros o plásticos termoformables, son los materiales que comúnmente se emplean en la fabricación de las partes de la ortesis que contienen la masa muscular de la pantorrilla y muslo, prácticamente a la medida.

Las ortesis se pueden clasificar de modo general en dos grupos según su principio de funcionamiento, pudiendo ser: pasivas o activas.

### 1.2.1 Ortesis pasivas

El concepto de ortesis pasiva de marcha se refiere al tipo de aparatos cuyo diseño contempla, únicamente partes mecánicas. La mayoría de los diseños existentes no permiten el movimiento en ninguna de las articulaciones del cuerpo y es necesario el uso de muletas o andaderas, pero actualmente se desarrollan sistemas en los que se logra brindar movilidad en las articulaciones sin utilizar ningún tipo de actuador, simplemente aprovechando aspectos físicos como la gravedad, el balanceo y elementos pasivos acumuladores de energía y transmisores de movimiento como resortes y cables. La ortesis pasiva de mayor uso es la denominada ortesis de marcha recíprocante (RGO, por sus siglas en inglés), que aprovecha la extensión de la cadera derecha, para inducir la abducción de la cadera izquierda y viceversa (ver figura 2).

Figura 2. Ortesis de marcha recíprocante (RGO)



Fuente: RSLSTEEPER, Riverside Orthopaedic Centre.

“Las ventajas que ofrecen estas ortesis son, un diseño relativamente sencillo, funcional, bajo costo, fácil de ponerse y quitarse y que se puede adaptar a diferentes tamaños y tallas de individuos sin grandes modificaciones, al diseño original. Un factor importante es la fácil disponibilidad de este tipo de ortesis. El punto débil de este desarrollo se concentra en la alta energía metabólica que el usuario consume al desplazarse usando dicha ortesis, lo que causa rápidamente su fatiga rápidamente. Sin embargo, le permite al sujeto sentarse, pararse, mantener una posición vertical y desarrollar una locomoción moderada con ayuda de muletas o de una andadera de apoyo; lo que representa cierto sentido de independencia. Además, permite a los sujetos interactuar familiarmente, socialmente y laboralmente. En este tipo de pacientes, mantenerse de pie de manera regular, aun algunos minutos al día, significa varios beneficios terapéuticos tales como; mejorar la circulación, un funcionamiento más eficiente de órganos internos, reducción en la tasa de descalcificación ósea y disminuir la incidencia de úlceras por presión” [ (VILLANUEVA D., 2001)].

### **1.2.2 Ortesis activas**

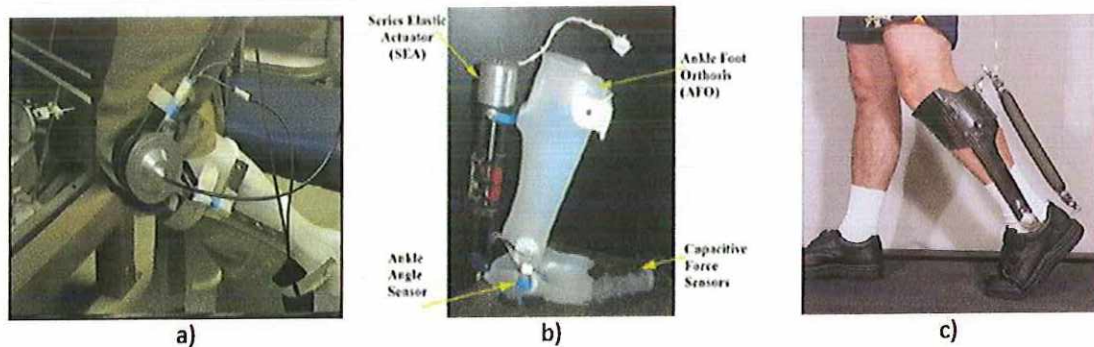
Las ortesis activas se han convertido hoy en día en una aplicación particular de los exoesqueletos, que consisten en un mecanismo estructural externo acoplado a la persona y cuyas juntas y eslabones corresponden a las de la parte del cuerpo humano que emula. El contacto entre el usuario y el exoesqueleto permite transferir potencia mecánica y señales de información. A diferencia de las ortesis pasivas, estos dispositivos son sistemas mecatrónicos que utilizan actuadores para proveer la fuerza y la movilidad de las articulaciones. Generalmente son controlados por una unidad central de procesamiento que tiene conectados diferentes instrumentos de medida (giroscopios, inclinómetros, sensores angulares, sensores mioeléctricos, entre otros) dependiendo del algoritmo de control que utilice [ (VILLANUEVA D., 2001)].

Cuando se desea o necesita reproducir o imitar los movimientos humanos, como es el caso de la locomoción artificial, una de las primeras cosas que se tienen que definir, es el tipo de actuador que se va a utilizar. Para ello es importante tener presente algunas de las características de los músculos humanos, entre las que destacan la densidad de energía ( $0.07 \text{ J/cm}^3$ ), su velocidad de respuesta ( $<100 \text{ ms}$ ), y su porcentaje de acortamiento o recorrido útil ( $> 40 \%$ ). Actualmente no hay disponible, actuador alguno que reúna todas las características del músculo humano, por lo tanto la tarea de la selección se debe hacer buscando cubrir las características o propiedades que mayor relevancia tengan, para la función específica. En el diseño de ortesis activas, se han empleado principalmente actuadores del tipo eléctrico, neumático, hidráulico, frenos magnéticos, dispositivos elásticos y resortes eléctricos [ (VILLANUEVA D., 2001)] (ver figura 3).

Los actuadores hidráulicos y neumáticos, son dispositivos capaces de transmitir grandes energías a las partes móviles, pero tienen la desventaja de ser de gran tamaño y de emplear un gran número de partes y accesorios asociados con su operación, además de requerir un mantenimiento continuo debido a problemas como condensación y fugas, lo que además involucra la seguridad del sujeto. Por otro lado los frenos y embragues electromagnéticos fabricados con partículas de las denominadas tierras raras, representan una buena alternativa, con ellos se han obtenido patrones de marcha uniformes, reproducibles y consistentes durante varias sesiones; son elementos limpios, silenciosos y de alta disponibilidad comercial; su principal desventaja es el peso y su tamaño, lo cual repercute en incrementos del costo metabólico y limitaciones en su uso cotidiano. Actualmente los actuadores que mayor uso tienen son los motores eléctricos en combinación con elementos acumuladores de energía como resortes y elásticos, debido a su fácil control y disponibilidad comercial [ (VILLANUEVA D., 2001)].

Entre las principales ventajas de las ortesis activas de marcha se puede mencionar que los patrones de marcha se vuelven uniformes al tener mayor control sobre las articulaciones y que el paciente puede hacer recorridos con una disminución del gasto cardiovascular, en comparación con ortesis pasivas. De esta manera, las áreas de oportunidad para desarrollos en el campo de las ortesis activas se enfocan en brindar mayor independencia de locomoción, reducir el costo metabólico asociado con su uso, reducir el peso y tamaño de los actuadores, el uso de nuevos materiales, y mejorar los programas de control para obtener patrones de marcha más estables y armoniosos [ (VILLANUEVA D., 2001)]. Además es importante incrementar la confiabilidad de los sistemas y reducir los costos para que sean comercialmente disponibles.

**Figura 3.** Ejemplos de actuadores en ortesis activas.



Fuente: (a) Ortesis de rodilla con actuador de resorte. (Tomado de S.Gharooni, M.O.Tokhi, B.Heller . *The use of elastic element in a hybrid orthosis for swing phase generation in orthotic gait*), (b) Ortesis de tobillo y pie con actuador serie eléctrico-elástico (Tomado de Blaya J, Herr H. *Adaptive Control of a Variable-Impedance Ankle-Foot Orthosis to Assist Drop-Foot Gait. IEEE Transactions on Neural Systems & Rehabilitation Engineering 2004*), (c) Ortesis de tobillo con músculo neumático (Tomado de Gordon KE, Sawicki GS and Ferris DP. *Mechanical performance of artificial pneumatic muscles to power an ankle-foot orthosis. Journal of Biomechanics, 2006*).

### 1.2.3 Trabajos relacionados

Los sistemas de exoesqueleto para operadores humanos ofrecen un amplio rango de aplicaciones. Como se mencionó en el apartado anterior, en pacientes con problemas físicos estos dispositivos permiten asistir las terapias de rehabilitación guiando los movimientos de las trayectorias correctas para ayudar al paciente a reaprender los patrones de motricidad y dar fuerza de soporte para realizar los movimientos, recibiendo en este caso particular el nombre de ortesis activa. Por otro lado, en ambientes industriales y militares los exoesqueletos se perfilan como un amplificador de potencia permitiendo al usuario soportar a través de la estructura mecánica grandes cargas por tiempos largos, evitando así lesiones y accidentes debido a sobreesfuerzos del trabajo físico. Aún más, dependiendo del tamaño, el peso y la ergonomía del dispositivo, los exoesqueletos pueden llegar a resultar beneficiosos en la vida cotidiana, especialmente para personas de la tercera edad. Los exoesqueletos también ofrecen una forma única de retroalimentar señales de fuerza al cuerpo humano, así que se pueden usar como interfaces sensoriales para teleoperación industrial, videojuegos y entretenimiento, o en monitoreo y entrenamiento, entre otras aplicaciones.

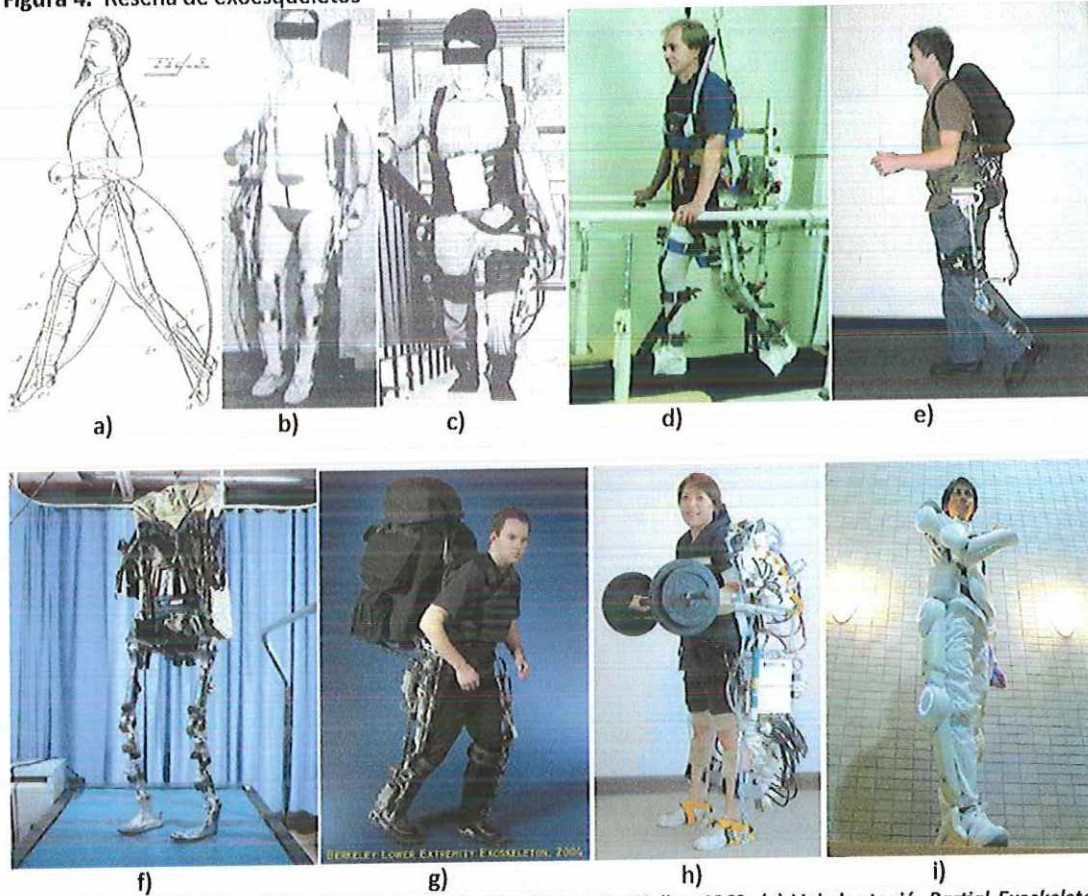
De este modo, siendo las aplicaciones de los exoesqueletos tan numerosas, muchos grupos de investigación han mostrado interés en este tópico, y especialmente en estos últimos años han surgido proyectos y publicaciones muy interesantes en el tema, lideradas principalmente por centros universitarios de investigación y por instituciones médicas y militares de algunos países. Según los reportes en el área de exoesqueletos activos, los pioneros son proyectos de Japón, Estados Unidos, Canadá, y algunos países europeos como Suiza, Italia, Francia y Alemania [ (GUIZZO, y otros, 2005)].

Básicamente, la documentación existente de investigaciones y proyectos realizados en esta temática se puede dividir en dos grupos según la aplicación final del exoesqueleto. Por un lado se encuentran los exoesqueletos como amplificadores de potencia, que es el principal enfoque de desarrollo concebido desde la ciencia ficción con su idea del súper soldado; y por otro lado se encuentran los exoesqueletos como tecnologías de asistencia para individuos con problemas de movilidad, que surgieron por el desarrollo de las ya mencionadas ortesis activas para los centros de rehabilitación. Esta última aplicación está apenas en desarrollo y no existe aún un dispositivo accesible a la mayoría que asista a los discapacitados en su vida cotidiana fuera de los centros terapéuticos. Se espera que con el avance de la tecnología, con la miniaturización de los mecanismos y con el creciente desarrollo de grupos en bioingeniería, se pueda lograr pronto este objetivo con un producto ergonómico.



La idea del exoesqueleto no es algo novedoso, y con el paso de la historia se han realizado numerosos diseños y prototipos, de los cuales se ilustran en la figura 4 algunos de los más relevantes para extremidades inferiores.

Figura 4. Reseña de exoesqueletos



Fuente: (a) Nicholas Yagn 1890, (b) Vukobratović. *The Kinematic Walker* 1969, (c) Vukobratović. *Partial Exoskeleton*. 1979, (d) Hocoma. *The robotic orthosis Lokomat*, (e) Walsh. *Under-actuated leg exoskeleton for load carrying augmentation* 2006, (f) S. Jacobsen. *Sarcos exoskeleton*, (g) H. Kazerooni. *BLEEX – The Berkley Lower Extremity Exoskeleton*. 2004, (h) Keijiro Yamamoto. *The Power Suit*. (i) Yoshiyuki Sankai. *HAL – Hybrid Assistive Limb*. 2002-2005.

Es importante señalar que actualmente los únicos dispositivos comercialmente disponibles son los referenciados en la figura 4.d y 4.i. El primero llamado Lokomat [ (HOCOMA)] es una unidad completa para centros de rehabilitación desarrollado por la empresa Hocoma y orientado principalmente a personas con lesión medular. El segundo llamado HAL es desarrollado por la empresa Cyberdine, está orientado para personas con debilidad muscular y aumento de potencia, su costo se aproxima a los US\$16.000 [ (Sankai)].

Ahora bien, a nivel latinoamericano es muy poca la documentación que se encuentra respecto al diseño y desarrollo de exoesqueletos orientados como ortesis activas, ya que la mayoría de los trabajos realizados se enfocan en la electro estimulación y en el desarrollo de prótesis y órtesis pasivas. Sólo se observan algunos títulos de algunas tesis de grado y proyectos de investigación en grupos de ingeniería mecatrónica, de ingeniería biomédica y de control, de algunas universidades e institutos, en especial en Brasil, Colombia, Cuba y Argentina. Específicamente a nivel nacional se encuentran referencias de títulos sobre exoesqueletos para rehabilitación, algunos para miembro superior y otros para miembro inferior, señalados en líneas de investigación de la Universidad Militar Nueva Granada, de la Escuela de Ingeniería de Antioquía, la Universidad Manuela Beltrán, y Universidad Autónoma de Bucaramanga, pero sin material descriptivo de los proyectos. Esto no implica que no haya otros grupos dedicados al tema, que por cuestiones de propiedad intelectual, no han publicado o se encuentran en etapa de desarrollo.

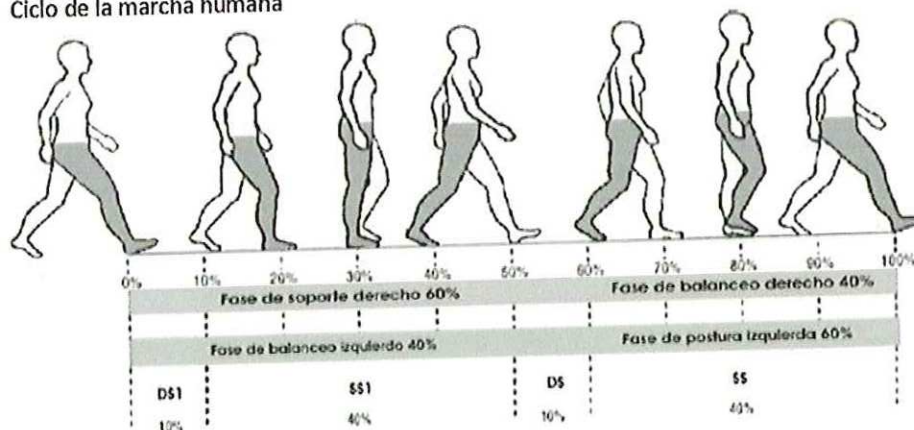
## 2. ASPECTOS GENERALES DE LA MARCHA HUMANA

A pesar de que la marcha humana es una actividad cotidiana que nos permite desplazarnos de un lugar a otro y nos brinda independencia y autonomía física para la realización de la mayor parte de nuestras tareas, el proceso de locomoción que la fundamenta es bastante complejo y particular por ser el resultado de un aprendizaje y un entrenamiento personal para lograr la bipedestación. Caminar es un proceso cíclico que conlleva interacciones coordinadas de las diferentes articulaciones y grupos musculares, por ello es posible dividirla en *fase de apoyo* y *fase de balanceo* para obtener un patrón de movimientos. Así pues, el propósito de este capítulo es presentar los aspectos relevantes del ciclo de la marcha, enfatizando en la cinemática angular de los miembros inferiores.

### 2.1 CICLO DE MARCHA

La marcha bípeda es una realización periódica de movimientos en cada uno de los miembros inferiores, alternando una fase de apoyo y una fase de balanceo (ver figura 5). La unidad fundamental de la marcha suele llamarse pasada o zancada y es un ciclo básico del caminado, definido como la secuencia de movimientos que ocurren entre dos repeticiones consecutivas de una configuración del cuerpo. Un paso tiene dos variables asociadas, el período (tiempo de inicio a fin) y la distancia recorrida en el ciclo. La velocidad promedio de la marcha oscila entre 75 y 140 pasos/min [ (CEPEDA, y otros, 2006)].

Figura 5. Ciclo de la marcha humana

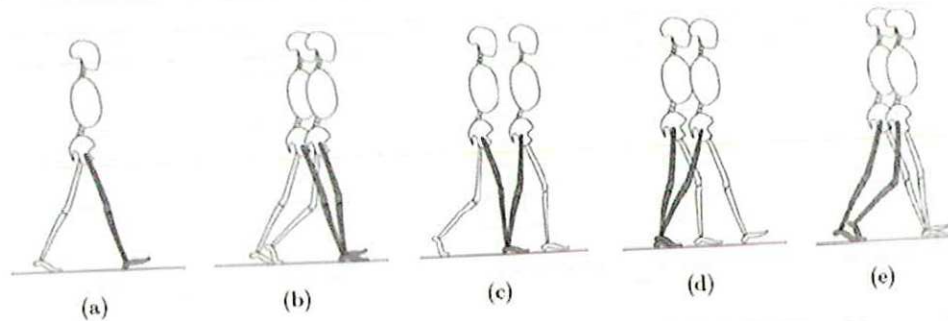


Fuente: Neumann, D.A. *Kinesiology of the muskoeskeletal system*.

### 2.1.1 Fase de apoyo

La fase de apoyo dura aproximadamente el 60% del paso completo y puede dividirse en tres etapas: la primera y la última se llaman *doble soporte* (DS) porque los dos pies hacen contacto simultáneo con el suelo. La subfase intermedia se denomina *soporte simple* (SS) ya que un sólo pie está en contacto con el suelo. En esta primera fase de apoyo ocurre una serie de eventos característicos entre las etapas mencionadas, como son: (a) *choque del talón* cuando el pie tiene contacto con el suelo por primera vez; (b) *apoyo plantar* que consiste en un intervalo durante el cual la planta del pie toma contacto pleno con el suelo y soporta todo el peso del cuerpo; (c) *apoyo medio* que es la parte media de la fase de soporte; el (d) *despegue del talón* donde la carga se transfiere de la parte posterior hacia la parte frontal del pie, y finalmente, (e) *despegue del pie*, momento en el cual el peso del cuerpo es transferido a la otra pierna y se finaliza la secuencia de la fase de apoyo (ver figura 6) [(PERRY, 1992)].

Figura 6. Eventos de la fase de apoyo

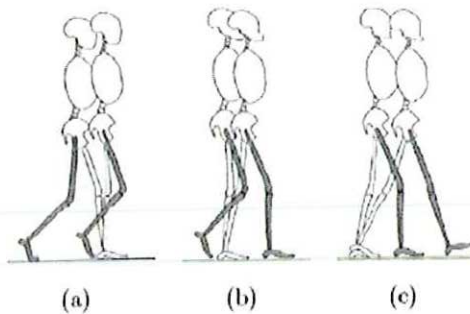


Fuente: Perry, J. (1992) *Gait Analysis: Normal and Pathological Function*. [(PERRY, 1992)]

### 2.1.2 Fase de balanceo

Aproximadamente, el 40% final del ciclo de la marcha es lo que se conoce como fase de balanceo, la cual se puede descomponer a su vez en tres eventos, como son: (a) *balanceo inicial*, que comienza al despegar el pie del suelo y finaliza cuando el muslo de esa pierna sobrepasa el muslo de la otra pierna, (b) *balanceo medio*, inicia desde el final de la aceleración y finaliza cuando la pantorrilla está vertical respecto al suelo, y (c) *balanceo final*, que inicia la desaceleración del miembro y finaliza con el contacto del talón en el suelo. Esta secuencia se observa en la figura 7 [(PERRY, 1992)].

Figura 7. Eventos de la fase de balanceo



Fuente: Perry, J. (1992) *Gait Analysis: Normal and Pathological Function*. (PERRY, 1992)

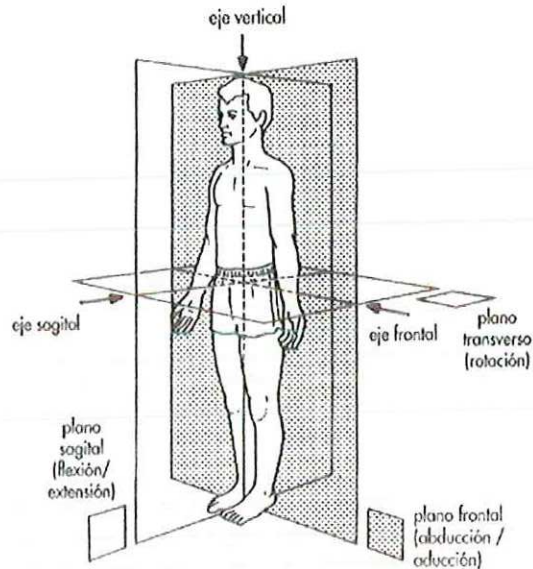
## 2.2 CINEMÁTICA

El análisis cinemático es el que nos permite describir el movimiento de los cuerpos sin tener en cuenta las causas que lo generan, sean fuerzas, momentos o energías. Por ello para poder generar y analizar patrones de marcha humana es relevante que se tenga en cuenta la cinemática de cada miembro involucrado.

Para una minuciosa descripción cuantitativa del movimiento de un cuerpo, se requiere un gran volumen de datos y un número elevado de operaciones y cálculos. Por ello, es necesario simplificar el análisis enfocándose solo en aquellas variables cinemáticas que mayor información ofrecen del movimiento [ (WINTER, 1979)]. En la descripción de la cinemática del ciclo de la marcha humana, suelen utilizarse tres conjuntos de parámetros: variables temporales (tiempo de duración de cada evento de la marcha), variables descriptivas (velocidad, aceleración, longitud del paso, cadencia, período), y parámetros cinemáticos articulares que son los más relevantes en el análisis de la marcha ya que permiten realizar estudios cinéticos para el cálculo de torques y de referencias para lazos de control.

La cinemática articular del miembro inferior describe los desplazamientos angulares de los segmentos corporales (muslo, pantorrilla y pie) alrededor de los ejes de movimiento (cadera, rodilla y tobillo respectivamente). Para este análisis cinemático del cuerpo humano, se utiliza una convención de marco de referencia (ver figura 8) para realizar las mediciones, el cual está formado por los planos: frontal, transversal y sagital.

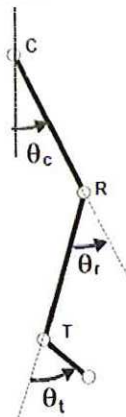
**Figura 8.** Planos de referencia para la cinemática humana



Fuente: Cuerpo consciente - Centro de actividades físicas para la salud. *Ejes y planos de movimiento del cuerpo.* Disponible en <http://metodopilates.blogspot.com/2006/03/ejes-y-planos-de-movimiento-del-cuerpo.html>

Este trabajo centra su análisis sobre el plano sagital por ser el plano que involucra mayor rango de movimiento articular del miembro inferior. En la figura 9 se ilustra la convención de ángulos relativos empleada, donde C, R y T son las articulaciones de cadera, rodilla y tobillo respectivamente.

**Figura 9.** Ángulos relativos entre los segmentos del miembro inferior

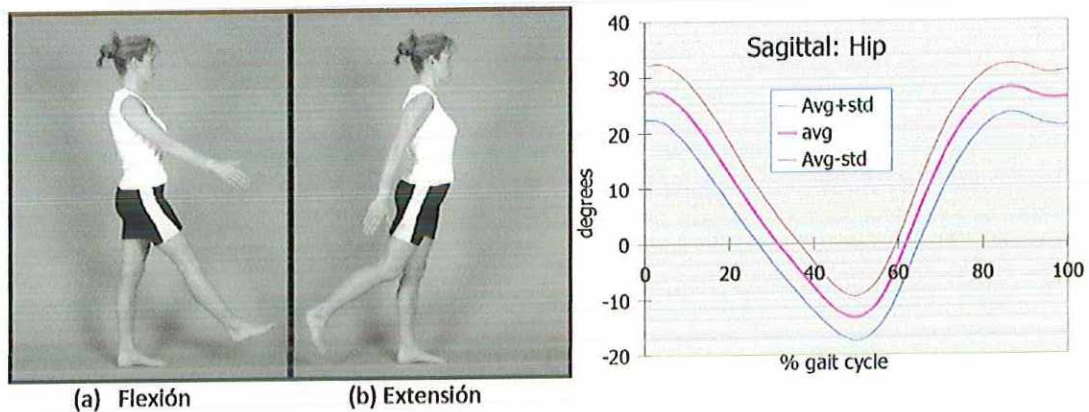


Fuente: El autor del proyecto

### 2.2.1 Cinemática angular de la cadera

El movimiento de la cadera tiene lugar en los tres planos. En el plano sagital se realizan los movimientos denominados flexión y extensión (ver figura 10), en el plano frontal se realiza la abducción y la aducción, y en el plano transversal se realiza la rotación externa y rotación interna.

Figura 10. Flexión y extensión de cadera durante la marcha.



Fuente: Stagni, Fantozzi, Capello. *Biomechanics of Human Movement*. 2006. Disponible en la página de internet <http://www.starter-project.com/Presentazioni/Cappello.pdf>

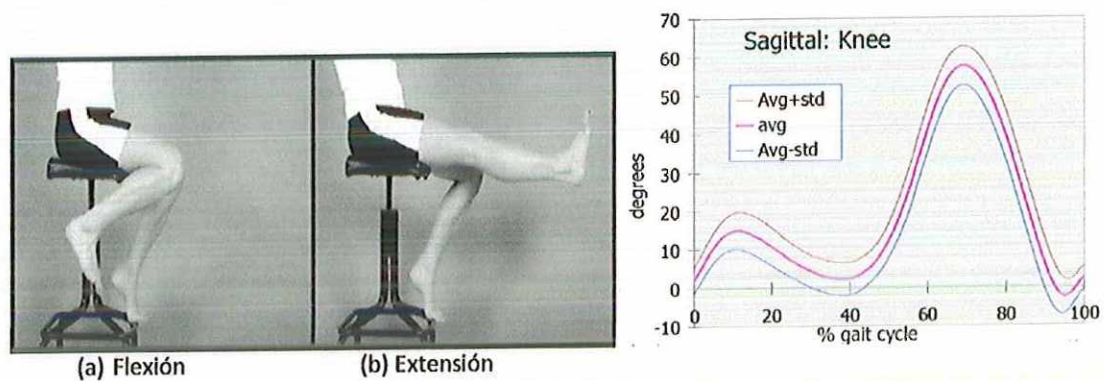
En el plano sagital el rango de flexión de la cadera es de 0 a 140° mientras que el rango de extensión es de 0 a 15°. Ahora bien durante la marcha no todo el rango de movimiento de la cadera es utilizado. En el momento de choque del talón, la flexión es aproximadamente de 30°, pero durante el despegue del talón se varía progresivamente hasta alcanzar un ángulo de extensión de 10°. En el despegue del pie y en la mayor parte del balanceo se alcanza un ángulo de flexión de 35°, para luego extender a 15° antes de chocar nuevamente el talón con el suelo. En la figura 10 se puede apreciar una curva común de la variación angular de la cadera en función del ciclo de la marcha [ (CEPEDA, y otros, 2006)].

### 2.2.2 Cinemática angular de la rodilla

La rodilla también presenta movimiento de flexión y extensión en el plano sagital (ver figura 11). El rango máximo entre el límite de la extensión y el de la flexión es de 0 a 140° aproximadamente.

En la marcha, durante el choque del talón, la rodilla está totalmente extendida. Luego en la parte media de la fase de apoyo se flexiona gradualmente hasta un ángulo de 20°. Al finalizar la fase de apoyo la rodilla vuelve a quedar extendida. En la fase de despegue del pie la rodilla se flexiona aproximadamente unos 40° y continua flexionándose alrededor de los 60° antes de iniciar la fase de aceleración en la mitad del balanceo. Posteriormente con la desaceleración se extiende nuevamente para prepararse al nuevo choque de talón. En la figura 11 se pueden observar más detalladamente estas variaciones [ (CEPEDA, y otros, 2006)].

**Figura 11.** Flexión y extensión de rodilla durante la marcha.



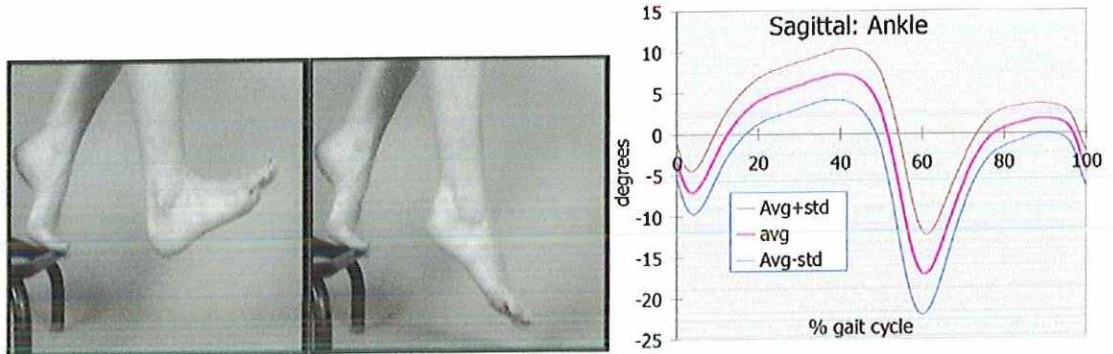
Fuente: Stagni, Fantozzi, Capello. *Biomechanics of Human Movement*. 2006. Disponible en la página de internet <http://www.starter-project.com/Presentazioni/Cappello.pdf>

### 2.2.3 Cinemática angular del tobillo

Dentro del plano sagital el tobillo también realiza movimiento de flexión y extensión, denominados particularmente flexión plantar y flexión dorsal. El rango normal en estado pasivo de la flexión dorsal alcanza un máximo entre los 10 a 20°, mientras que la flexión plantar tiene un valor máximo entre los 40 y 55° de manera pasiva. En la figura 12 se puede observar las alternancias entre la flexión plantar y la dorsal durante la marcha [ (CEPEDA, y otros, 2006)].



**Figura 12.** Flexión dorsal y flexión plantar del tobillo durante la marcha.



Fuente: Stagni, Fantozzi, Capello. *Biomechanics of Human Movement*. 2006. Disponible en la página de internet <http://www.starter-project.com/Presentazioni/Cappello.pdf>

## 2.3 TÉCNICAS PARA CAPTURA DE MOVIMIENTO CORPORAL

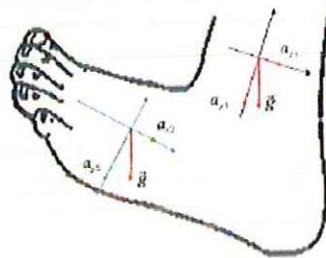
Las técnicas para análisis de movimientos y posturas del cuerpo humano no es algo novedoso, de tal manera que todos estos años de desarrollo tecnológico han resultado en una gran variedad de sistemas útiles para medir la posición de los segmentos corporales y los ángulos relativos entre ellos. Estos sistemas se pueden categorizar según su principio de funcionamiento en: mecánicos, ópticos, magnéticos, acústicos e inerciales [ (WINTER, 1979)]. A continuación se describen las técnicas más utilizadas en el análisis de la marcha humana:

### 2.3.1 Acelerometría

El principal uso de los acelerómetros y giroscopios, se centra en aplicaciones de análisis cinético, ya que como su nombre lo indica, son dispositivos que permiten medir la aceleración e inclinación de los cuerpos, lo cual es punto de partida para realizar cálculos de variables de fuerza, torques, velocidad y posición, con solo aplicar algunas relaciones matemáticas. Su principio se fundamenta en la segunda *ley de Newton* y en la *ley de Hooke*, ya que principalmente son transductores de fuerza que generan una respuesta en relación a las reacciones asociadas a una determinada aceleración. El esquema más simple consta de un sistema masa resorte. Los acelerómetros pueden venir con uno, dos o tres ejes de medida.

Para realizar mediciones de ángulos articulares con acelerómetros, es necesario que se coloque un acelerómetro biaxial en cada eslabón que tiene relación con la articulación y lo más cercano a ella, de manera que los dos ejes efectivos del sensor definan un sistema cartesiano de referencia (ver figura 13). Las aceleraciones soportadas por los ejes efectivos de cada acelerómetro corresponden a las componentes de un mismo vector en dos sistemas de referencia diferentes y pueden relacionarse mediante transformaciones homogéneas de coordenadas para obtener así la rotación existente [ (CEPEDA, y otros, 2006)].

Figura 13. Ubicación esquemática de acelerómetros para medir rotación articular



Fuente: R. Gómez, C. Ochoa. *Diseño de un equipo prototipo para la medición de ángulos articulares del miembro inferior en el plano sagital utilizando acelerómetros*. 2006.

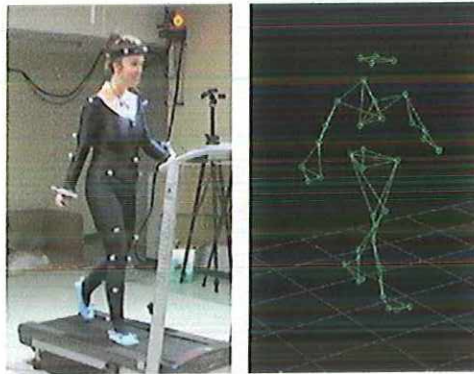
La principal ventaja de esta técnica es que la señal de salida de los acelerómetros está disponible inmediatamente para ser grabada o digitalizada para su procesamiento en un computador. Como desventajas se pueden mencionar: la aceleración es relativa a la ubicación del sensor en el segmento corporal, el costo de los acelerómetros puede ser excesivo si se requiere un gran número de ellos, los requerimientos de procesamiento para la conversión análogo digital son altos ya que por cada articulación se requieren dos acelerómetros que equivalen a 4 señales para digitalizar, muchos acelerómetros son altamente sensibles a ruidos o a daños físicos.

### 2.3.2 Análisis por video y fotografía

En el procesamiento de imágenes suelen emplearse varios tipos de sistemas, de los cuales los más importantes son las cámaras de cine y televisión, las exposiciones múltiples de imágenes, y los sistemas optoelectrónicos [ (WINTER, 1979)].

Estas técnicas cuentan con la ventaja de no necesitar un contacto directo con el paciente, de modo que el sujeto puede caminar libremente, con lo que se obtienen resultados más cercanos a la realidad. Para facilitar el procesamiento de la imagen suelen emplearse trajes y marquillas especiales (figura 14).

**Figura 14.** Sistema óptico para captura de movimiento



Fuente: Laboratorio de Realidad Virtual de la Escuela Politécnica Federal de Lausanne. Disponible en [http://vrlab.epfl.ch/research/LO\\_locomotin\\_engine.html](http://vrlab.epfl.ch/research/LO_locomotin_engine.html)

La principal desventaja de estos sistemas es que se requiere de un ambiente controlado, con ciertas condiciones de luz y color de fondo, se requiere una correcta disposición espacial de las cámaras que graban el movimiento y el costo de instalar un laboratorio de análisis de movimiento por video y fotografía es bastante alto.

### 2.3.3 Electrogoniometría

La electrogoniometría es una herramienta que sirve para medir la variación de los ángulos articulares por medio de un sistema electromecánico (ver figura 15), cuya parte mecánica es la estructura rotacional y los eslabones de acople, y la parte eléctrica es un transductor resistivo de respuesta lineal. De esta manera la cinemática angular se mide en relación a la variación de tensión que presenta el transductor entre un terminal móvil y uno de referencia.

Entre las ventajas de estos dispositivos se tiene que son de bajo costo, permiten obtener la señal de salida en tiempo real para grabación o procesamiento, la señal de rotación es independiente del plano de movimiento de la articulación.

Como desventajas se puede mencionar que los datos angulares son relativos al sistema de referencia local de la articulación y no de un sistema absoluto que a veces es preferido para convenciones de procesamiento, puede requerir un gasto de tiempo considerable para su ajuste y alineamiento, y el peso y complejidad del mecanismo puede limitar la libertad de movimiento del paciente [ (WINTER, 1979)].

**Figura 15.** Análisis de marcha usando un electrogoniómetro



Fuente: C. Kirtley. *History of the study of locomotion*. Disponible en <http://www.univie.ac.at/cga/history/ww2.html>

### **3. SISTEMA DE MONITOREO Y ASISTENCIA DE LA MARCHA (LEGXOS)**

Cada año la robótica está fortaleciendo su papel en la neuro-rehabilitación de pacientes que sufren de lesiones al sistema nervioso, quienes desarrollan trastornos del movimiento debido a discapacidades sensoriales y motrices, tales como debilidad muscular, pérdida de coordinación muscular, espasticidad y otros síntomas como el aumento de tono muscular y los reflejos exagerados [ (SCHIELE, y otros, 2006)]. La consiguiente incapacidad para realizar las funciones elementales con sus extremidades, tales como caminar, hace imposible la autonomía en su vida cotidiana.

Ante esta situación, un tratamiento que tiene como objetivo fomentar una mejor y más rápida recuperación del paciente es la *terapia con movimiento intensivo*, la cual mejora las funciones y las capacidades motrices en el paciente, por la estimulación neuronal con ejercicios de movimientos repetitivos, ya que así se reactivan los caminos y conexiones entre neuronas. Otro objetivo de esta terapia es extender los tejidos blandos con el fin de aumentar el rango pasivo del movimiento e incrementar la fuerza muscular. La eficiencia del tratamiento se mejora significativamente con sesiones de larga duración y si el tratamiento inicia lo más pronto posible después de que aparece la discapacidad e involucra entrenamiento con movimientos repetitivos.

La calidad del movimiento inducido es crucial para la recuperación. Para una rehabilitación de pierna es esencial una combinación determinada de señales de entrada para generar los patrones de locomoción en la parte baja de la columna vertebral. Esto lleva a la conclusión que una terapia es mucho más eficiente si el movimiento inducido es cíclico, reproducible, rítmico y fisiológico.

Actualmente, las terapias de movimiento son en su mayoría implementadas manualmente por un fisioterapeuta, pero éste es afectado por la fatiga, ya que debe ejecutar el tratamiento en posiciones no favorables desde el punto de vista ergonómico. Esto trae consigo que los movimientos inducidos sean inconsistentes cuando el cansancio afecte al terapeuta, lo cual está en detrimento del paciente. Las terapias actuales son generalmente de corta duración y hasta suelen omitirse si el paciente presenta síntomas de espasmo severo.

Aquí es donde la terapia manual tiene sus límites y la terapia con aplicaciones robóticas muestra sus ventajas. Los robots son ciertamente diseñados para ejecutar de manera precisa tareas repetitivas que requieren potencia mecánica, por tanto se muestran muy útiles para agilizar la rehabilitación de pacientes con problemas motrices, ya que son flexibles para programar las terapias con la exactitud del movimiento a inducir y con el número de repeticiones deseadas, lo cual permite prolongar las sesiones de entrenamiento pues no entra en juego el factor fatiga del terapeuta.

A continuación se describe de manera general la propuesta específica de este proyecto, la cual involucra un sistema mecatrónico que aprovecha la simetría del sistema músculo-esquelético de las extremidades inferiores y de los movimientos que éstas ejecutan, con el fin de monitorear y asistir la marcha en sesiones de rehabilitación de personas con dificultades para realizar movimientos con sus piernas. Se señalan las especificaciones del sistema, los bloques que lo componen y su esquema de funcionamiento.

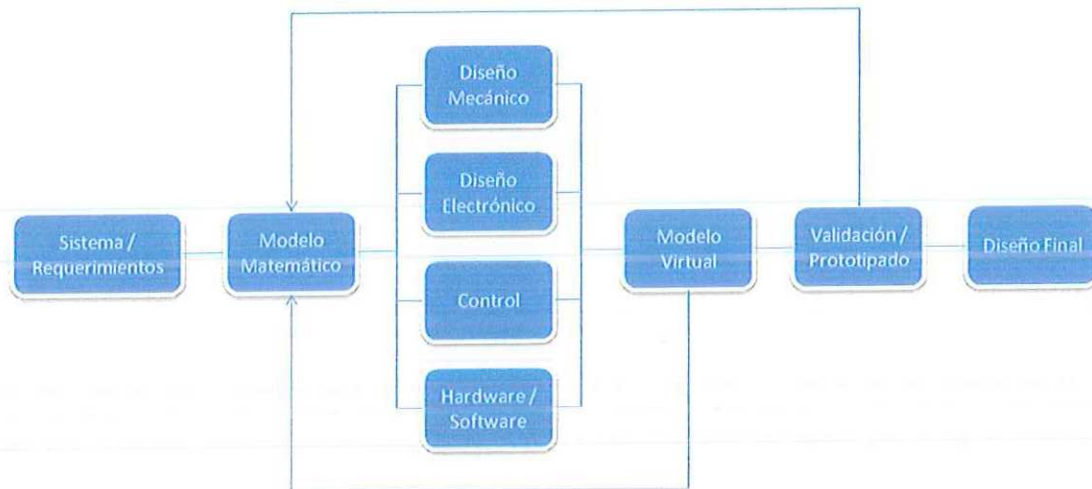
### **3.1 REQUERIMIENTOS Y ESPECIFICACIONES**

Los requerimientos mínimos que se establecieron para el diseño del sistema son:

- Poder registrar el movimiento ejecutado en una pierna sana, específicamente la rotación del muslo y la pantorrilla dentro del plano sagital, en referencia a las juntas de la cadera y rodilla respectivamente.
- Reproducir el movimiento capturado de la pierna sana en un entorno gráfico de validación para luego inducirlo en un sistema esclavo.
- El sistema debe ser flexible y permitir cambios en el patrón de movimientos para lograr un entrenamiento eficiente.
- El hardware y software deben presentar fácil interacción con el usuario.

Para cumplir con lo anterior se adoptó el proceso de diseño mecatrónico, ya que en contraste con otras metodologías de diseño, éste integra diferentes dominios de la ingeniería con el fin de aprovechar las relaciones existentes entre las diversas perspectivas de diseño para optimizar así un modelo representativo del sistema en interés. La clave principal para dar flexibilidad y robustez a este proceso de diseño es el modelamiento sinérgico y la simulación de los sistemas, que permiten evaluar las respuestas ante diversos criterios y condiciones de funcionamiento. En la figura 16 se puede observar un esquema general del proceso adoptado y se puede evidenciar que en la etapa de diseño, las líneas de unión del diagrama intencionalmente no presentan dirección para enfatizar que las diferentes perspectivas se trabajan en paralelo y se interrelacionan unas con otras a fin de optimizar el diseño final.

Figura 16. Esquema general del proceso de diseño mecatrónico adoptado



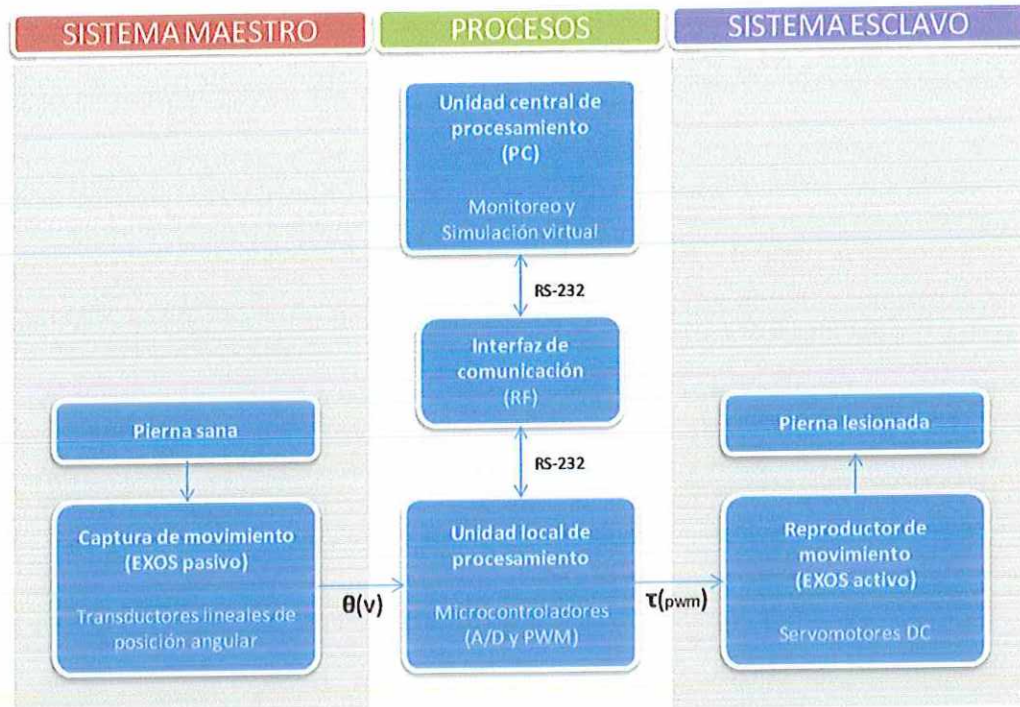
Fuente: El autor del proyecto

Como resultado de este proceso, la solución presentada en este proyecto busca integrar tres áreas tecnológicas de reciente fortalecimiento como son: la robótica, la teleoperación y el desarrollo de entornos virtuales, no sólo con el fin de cumplir los requerimientos estipulados sino también con la intención de lograr un mayor acercamiento por parte del autor a dichas áreas que se perfilan como herramientas importantes para lograr aplicaciones de gran interés dentro del campo de la ingeniería biomédica.

El sistema desarrollado se puede dividir en tres subsistemas de acuerdo a las acciones que se llevan a cabo en cada uno de ellos (ver figura 17). A continuación se describe de manera general el funcionamiento de cada uno de ellos:

- *Sistema Maestro*, es el encargado de generar los movimientos articulares deseados a reproducir en la pierna lesionada. Está conformado por la pierna sana y por un exoesqueleto pasivo acoplado a ella, similar a un electrogoniómetro, que permite monitorear la cinemática articular de los movimientos de la pierna dentro del plano sagital. La variación angular es captada a través de potenciómetros rotacionales de respuesta lineal que se encuentran acoplados dentro de las articulaciones del mecanismo.
- *Sistema Esclavo*, es el encargado de reproducir las trayectorias de movimiento registradas por el sistema maestro. Está conformado por la pierna afectada y por un exoesqueleto activo acoplado a ella, encargado de aplicar los torques necesarios en las juntas de la pierna a través del accionamiento y control de posición de servomotores de corriente continua.

**Figura 17.** Diagrama de bloques del sistema de monitoreo y asistencia de la marcha.



Fuente: El autor del proyecto

- *Unidad de Procesos*, es la encargada del procesamiento de datos a nivel central y a nivel local, así como de establecer la comunicación entre el sistema maestro y el sistema esclavo. En esta unidad se diferencian tres bloques particulares que son: la *unidad central de procesamiento*, la *unidad de procesamiento local* y la *interfaz de comunicación*.
- La unidad de procesamiento local está conformada por dos etapas: una primera tarjeta desarrollada para la adquisición de señales analógicas encargada de muestrear a una frecuencia determinada los niveles de tensión de los potenciómetros relativos a las articulaciones del sistema maestro y convertirlas en señales digitales de 8 bits para su posterior procesamiento; y por otro lado una tarjeta desarrollada para el accionamiento de los actuadores, la cual convierte ordenes de posición en una modulación del ancho de pulso de la señal de control de cada servomotor. Ambas tarjetas manejan comunicación por protocolo RS-232.



- La unidad central de procesamiento está basada en un computador con plataforma Windows XP o Windows Vista. En esta unidad se ejecutan las aplicaciones de interfaz gráfica de usuario que involucran programación de alto nivel desarrolladas en el software MATLAB de Mathworks y LabVIEW de National Instruments. Los procesos que se realizan incluyen: (a) el control del muestreo y la recepción de los paquetes de bytes enviados por la unidad de procesamiento local relativos a las posiciones angulares del sistema maestro; (b) el procesamiento de dichos paquetes para obtener un valor numérico de los ángulos y graficarlos en tiempo real; (c) la animación del movimiento sobre un entorno gráfico 3D; (d) el cálculo de la dinámica para la activación y control de los actuadores del sistema esclavo; (e) el manejo de una base de datos de pacientes y el registro de históricos de las sesiones de terapia realizadas; (f) la comunicación serial con las otras unidades empleando el protocolo RS-232.
- La interfaz de comunicación es la encargada de realizar el transporte de datos digitales entre la unidad central de procesos y las unidades locales de cada subsistema. Está basada en comunicación serial inalámbrica por módulos de radio frecuencia para dar mayor libertad al usuario del exoesqueleto en la realización de los movimientos.

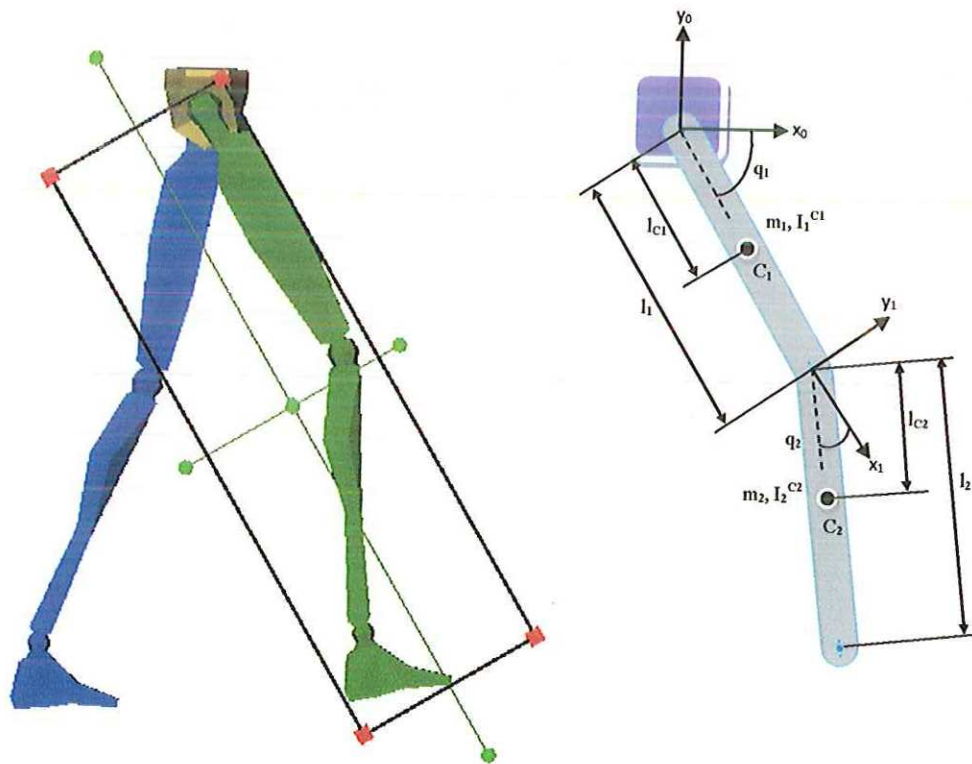
Es importante aclarar en este momento, que el alcance del proyecto no incluye la construcción final de la estructura real propuesta debido a que es un proyecto de investigación sin financiamiento externo que está enfocado principalmente en el diseño y simulación del sistema mecatrónico. Aún así, se presenta el diseño de un modelo funcional construido para validar el funcionamiento de cada una de las etapas que involucran el sistema completo. Específicamente se desarrolló una plataforma bípeda que permitiera evaluar el tema central de este trabajo que es la reproducción en un sistema esclavo del movimiento ejecutado por un sistema maestro aplicado a la rehabilitación de piernas.

La descripción ampliada de cada unidad del sistema se puede encontrar en los capítulos siguientes referentes al diseño mecánico, diseño electrónico y desarrollo de software.

## 4. MODELAMIENTO DEL SISTEMA

En este capítulo se presentan los diferentes modelos matemáticos y computacionales que se desarrollaron en este trabajo con el fin de describir la dinámica del movimiento de las piernas durante la marcha. Inicialmente sólo se modela una pierna en la fase oscilación asumiéndola como un doble péndulo con pivote en la cadera (ver figura 18), con el fin de comparar la funcionalidad de las estrategias de modelamiento empleadas y evaluar la certeza de los planteamientos. Es necesario el empleo de datos antropométricos y cinemáticos a fin de calcular ciertos parámetros físicos necesarios para la resolución de los modelos, para ello se emplearán los datos referidos por Winter.

Figura 18. Modelo de una pierna con dos grados de libertad.



Fuente: El autor del proyecto

En la figura 18 se señalan las variables geométricas e inerciales a incluir en los análisis dinámicos de la pierna, y se asignan los sistemas de referencia a emplear. A continuación se define cada variable involucrada:

$\{x_0, y_0\}$  : sistema de referencia de la cadera.

$\{x_1, y_1\}$  : sistema de referencia de la rodilla.

$C_1$  y  $C_2$  : centros de masa del muslo y de la pantorrilla respectivamente.

$m_1$  y  $m_2$  : masas del muslo y de la pantorrilla respectivamente.

$I_1^{C_1}$  y  $I_2^{C_2}$  : tensores de inercia del muslo y la pantorrilla respectivamente.

$l_1$  y  $l_2$  : longitud del muslo y de la pantorrilla respectivamente.

$l_{C_1}$  : distancia medida desde la cadera al centro de masa del muslo.

$l_{C_2}$  : distancia medida desde la rodilla al centro de masa de la pantorrilla.

$q_1$  : ángulo entre la horizontal y el eje longitudinal del muslo.

$q_2$  : ángulo entre la proyección longitudinal del muslo y la pantorrilla.

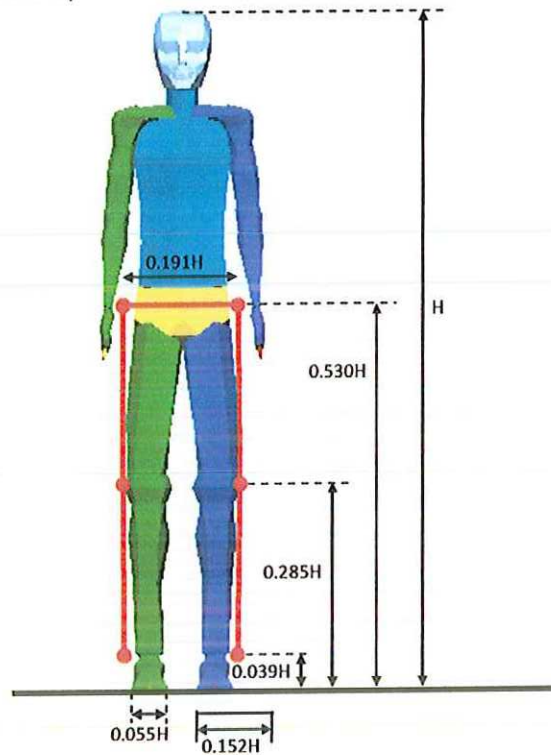
## 4.1 DATOS ANTROPOMÉTRICOS

La antropometría es la ciencia encargada de realizar mediciones físicas del cuerpo humano con el fin de determinar diferencias entre individuos y grupos. Por ello, el uso de datos antropométricos es relevante en este trabajo, pues se trata de una interfaz hombre-máquina que permite intercambiar señales y potencia entre el mecanismo propuesto y las piernas de la persona, lo que hace necesario tener conocimiento de longitudes, masas, inercias y localizaciones del centro de masas de los cuerpos, con el fin de modelar y dimensionar correctamente las características del sistema.

### 4.1.1 Longitud de los segmentos

En este trabajo se emplearán las relaciones establecidas por Drillis y Contini [ (WINTER, 1979)], donde se expresan las longitudes de los segmentos como una fracción de la altura total del individuo. En la figura 19 se ilustran las aproximaciones empleadas para la pierna.

Figura 19. Longitud de los segmentos de la pierna en función de la altura total.



Fuente: El autor del proyecto.

En estos análisis, los cálculos se basan en la contextura y dimensiones del autor. Por tanto, teniendo como base una altura  $H$  máxima de 1.75 m, y la nomenclatura empleada en la figura 18, las longitudes de los segmentos de la pierna a modelar son:

$$l_1 = (0.530 - 0.285)H = 0.245 \times 1.75 = 0.4287 \text{ m}$$
$$l_2 = (0.285 - 0.039)H = 0.246 \times 1.75 = 0.4305 \text{ m}$$

#### 4.1.2 Masa y centro de masas

El *centro de masas* de un cuerpo es aquel punto en el espacio que se comporta dinámicamente como si estuviera sometido a la resultante de las fuerzas del sistema. Es aquel punto en el cual el cuerpo se puede modelar como una masa puntual concentrada.

De igual manera que para las longitudes de los segmentos, para la masa y la ubicación del centro de masas existen tablas antropométricas donde se resumen estas propiedades para cada segmento como una fracción de la masa total del individuo y de su altura. A continuación se tabulan las relaciones específicas aplicadas a nuestro sistema de pierna formado por dos eslabones:

**Tabla 1.** Datos antropométricos para miembro inferior

Segmento	Peso de segmento/ peso total	Centro de masa/ longitud de segmento		Radio de giro/ longitud de segmento		
		Proximal	Distal	Centro de Gravedad	Proximal	Distal
Pie	0.0145	0.50	0.50	0.475	0.690	0.690
Pantorrilla	0.0465	0.433	0.567	0.302	0.528	0.643
Muslo	0.100	0.433	0.567	0.323	0.540	0.653
Pie y pantorrilla	0.061	0.606	0.394	0.416	0.735	0.572
Total pierna	0.161	0.447	0.553	0.326	0.560	0.650

Fuente: Winter, David. *Biomechanics and motor control of human movement*. (1990)

El *tensor de inercia* es una generalización del momento de inercia de una masa puntual y representa la distribución de la masa de un cuerpo en torno a los ejes de giro, y la resistencia que presenta el cuerpo a un cambio en la velocidad angular. Se denota por  $I$  y su valor depende del punto en el que la rotación tenga lugar, siendo mínimo cuando dicho punto se ubica en el centro de masas del cuerpo.

Matemáticamente está representado por una matriz simétrica de tamaño 3x3, donde las componentes de este tensor de inercia en una base ortonormal XYZ pueden calcularse a partir de los tres momentos de inercia según esos tres ejes perpendiculares. Así:

$${}^A I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}$$

donde:

$$I_{xx} = \int_V \rho (y^2 + z^2) dx dy dz$$

$$I_{yy} = \int_V \rho (z^2 + x^2) dx dy dz$$

$$I_{zz} = \int_V \rho (x^2 + y^2) dx dy dz$$

$$I_{xy} = I_{yx} = \int_V -\rho xy \, dx dy dz$$

$$I_{yz} = I_{zy} = \int_V -\rho yz \, dx dy dz$$

$$I_{zx} = I_{xz} = \int_V -\rho zx \, dx dy dz$$

Ahora bien, si se ubica el sistema de referencia en el centro de masas del cuerpo, entonces el tensor de inercia es mínimo y se reduce a:

$${}^{CM}I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix}$$

Con esta información se pueden calcular entonces los valores de interés para el modelo del exoesqueleto de pierna con dos grados de libertad, teniendo en cuenta que el peso total de la persona analizada es de 65 kg y que los segmentos de la pierna (muslo y pantorrilla), se simplifican geoméricamente a cilindros sólidos, así:

Masas:

$$m_1 = 0.100 \times 65 = 6.5 \text{ kg}$$

$$m_2 = 0.0465 \times 65 = 3.0225 \text{ kg}$$

Centro de masas:

$$CM_1 = 0.433 \times 0.4287 = 0.1856 \text{ m}$$

$$CM_2 = 0.433 \times 0.4305 = 0.1864 \text{ m}$$

Tensores de inercia:

$${}^{CM_1}I_1 = \begin{bmatrix} \frac{1}{4}m_1r_1^2 + \frac{1}{12}m_1h_1^2 & 0 & 0 \\ 0 & \frac{1}{4}m_1r_1^2 + \frac{1}{12}m_1h_1^2 & 0 \\ 0 & 0 & \frac{1}{2}m_1r_1^2 \end{bmatrix}$$

$${}^{CM_1}I_1 = \begin{bmatrix} 0.1077 & 0 & 0 \\ 0 & 0.1077 & 0 \\ 0 & 0 & 0.0163 \end{bmatrix} \text{ kg} \cdot \text{m}^2$$

$${}^{CM_2}I_2 = \begin{bmatrix} \frac{1}{4}m_2r_2^2 + \frac{1}{12}m_2h_2^2 & 0 & 0 \\ 0 & \frac{1}{4}m_2r_2^2 + \frac{1}{12}m_2h_2^2 & 0 \\ 0 & 0 & \frac{1}{2}m_2r_2^2 \end{bmatrix}$$

$${}^{CM_2}I_2 = \begin{bmatrix} 0.0492 & 0 & 0 \\ 0 & 0.0492 & 0 \\ 0 & 0 & 0.0051 \end{bmatrix} \text{ kg} \cdot \text{m}^2$$

## 4.2 DINÁMICA

A diferencia de la cinemática, la dinámica no sólo estudia el movimiento de los cuerpos sino también las fuerzas que lo generan. Desde esa perspectiva, el problema dinámico en robótica estudia la relación entre el movimiento del sistema y los pares o fuerzas ejercidas por los actuadores de las articulaciones.

En dinámica son posibles dos problemas, el directo y el inverso. El problema *dinámico directo* consiste en calcular el movimiento del extremo del robot a partir de los pares y fuerzas ejercidos en las articulaciones. El problema *dinámico inverso* consiste en calcular los valores de los pares y fuerzas necesarias para lograr las posiciones, velocidades y aceleraciones de un movimiento deseado. Este último caso es el que se desarrolla en este trabajo ya que el sistema esclavo debe reproducir una trayectoria articular generada por el sistema maestro.

Un buen modelado del problema dinámico inverso determina la eficiencia y rapidez de respuesta de los sistemas, por ello existen numerosas estrategias de modelamiento dinámico que buscan reducir los costes computacionales en términos de operaciones requeridas para encontrar la solución. Entre los métodos de solución de la dinámica de sistemas de cuerpos rígidos se encuentran los *métodos iterativos* que resuelven las ecuaciones generales de forma numérica, las *soluciones cerradas* que resultan de obtener expresiones analíticas del problema dinámico para una configuración de cuerpos específica, los *métodos recursivos* y actualmente métodos computacionales de análisis de sistemas mecánicos cuyo modelamiento es gráfico.

A continuación se presentan tres desarrollos para modelar el movimiento del exoesqueleto de una pierna, con el fin de determinar qué técnica de modelado es más eficiente para luego simular el sistema completo de ambas piernas. Se asumió que todo el cuerpo de la persona está estático excepto la pierna en cuestión, que tiene como pivote la juntura de la cadera, y que se ha simplificado a un robot formado por una cadena cinemática de 2 grados de libertad (ver figura 18).

#### 4.2.1 Formulación Lagrangiana

Esta formulación permite describir la dinámica del sistema a partir de un balance de energía, de tal manera que el exoesqueleto de pierna se considera como una caja negra y se tiene en cuenta sólo la energía almacenada, que se expresa en términos de energía cinética y potencial.

Específicamente el Lagrangiano es una función escalar que se define como la diferencia entre las energías cinética y potencial de un sistema mecánico, en función de las llamadas coordenadas generalizadas que en el caso del exoesqueleto son las variables articulares.

$$\mathcal{L}(q, \dot{q}) = E_C(q, \dot{q}) - E_P(q) \quad (4.1)$$

donde  $\mathcal{L}$  es el Lagrangiano,  $E_C$  la energía cinética y  $E_P$  la energía potencial.

A partir de la ecuación 4.1 se puede definir las ecuaciones del movimiento para el sistema:

$$\frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{q}} \right) - \frac{\partial \mathcal{L}}{\partial q} = \tau \quad (4.2)$$

donde  $\tau$  es un vector  $n \times 1$  formado por los pares de los motores que actúan sobre las articulaciones rotacionales del sistema.

Teniendo en cuenta la ecuación 4.1, se obtiene para la expresión 4.2 lo siguiente:

$$\frac{d}{dt} \left( \frac{\partial E_C}{\partial \dot{q}} \right) - \frac{\partial E_C}{\partial q} + \frac{\partial E_P}{\partial q} = \tau \quad (4.3)$$

La **energía cinética** total del exoesqueleto es la suma de las energías cinéticas de los  $n$  eslabones que integran la cadena cinemática:



$$E_C = \sum_{i=1}^n E_{C_i} \quad (4.4)$$

donde la energía cinética de un eslabón  $i$  es la suma de dos términos, el primero debido a la velocidad lineal, y el segundo a la velocidad angular del eslabón:

$$E_{C_i} = \frac{1}{2} m_i (\mathbf{v}_{C_i}^C)^T (\mathbf{v}_{C_i}^C) + \frac{1}{2} (\boldsymbol{\omega}_i^C)^T \mathbf{I}_i^C \boldsymbol{\omega}_i^C \quad (4.5)$$

donde:

$m_i$  = masa del eslabón  $i$ .

$\mathbf{v}_{C_i}^C$  = velocidad lineal del centro de masas del eslabón  $i$  con relación al sistema de referencia situado en el centro de masas.

$\boldsymbol{\omega}_i^C$  = velocidad angular del eslabón  $i$  con relación al sistema de referencia situado en el centro de masas.

$\mathbf{I}_i^C$  = tensor de inercia del eslabón  $i$  respecto al sistema de referencia situado en el centro de masas.

Cabe destacar que la velocidad angular es la misma para todos los puntos de un eslabón, pero no se puede decir lo mismo para la velocidad lineal, que resulta distinta en cada punto del eslabón; por ello se elige el sistema de referencia en el centro de masas del eslabón a fin de facilitar el cálculo de la energía cinética. El tensor de inercia también puede ser expresado respecto al mismo sistema de referencia.

Teniendo en cuenta que las velocidades lineales y angulares de la ecuación 4.5 son función de las variables articulares  $\mathbf{q}$  y de las velocidades articulares  $\dot{\mathbf{q}}$ , se puede expresar entonces la energía cinética en términos de las jacobianas de las coordenadas generalizadas,

$$E_C = \frac{1}{2} \dot{\mathbf{q}}^T \sum_{i=1}^n \left[ m_i \cdot \mathbf{J}_{\mathbf{v}_{C_i}}^T(\mathbf{q}) \cdot \mathbf{J}_{\mathbf{v}_{C_i}}(\mathbf{q}) + \mathbf{J}_{\boldsymbol{\omega}_i}^T(\mathbf{q}) \cdot \mathbf{I}_i^C \cdot \mathbf{J}_{\boldsymbol{\omega}_i}(\mathbf{q}) \right] \dot{\mathbf{q}}$$

$$E_C = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{D}(\mathbf{q}) \dot{\mathbf{q}} \quad (4.6)$$

Ahora bien, la **energía potencial** total almacenada en el exoesqueleto  $E_p$  equivale a la suma de las energías potenciales almacenadas por cada uno de los eslabones que lo conforman:

$$E_P = \sum_{i=1}^n E_{P_i} = E_P(\mathbf{q}) \quad (4.7)$$

donde la energía potencial de cada eslabón es:

$$E_{P_i} = -m_i \mathbf{g}^T \mathbf{p}_{C_i}^M \quad (4.8)$$

donde:

$m_i$  = masa del eslabón  $i$ .

$\mathbf{g}$  = vector de gravedad.

$\mathbf{p}_{C_i}^M$  = es el vector que localiza el centro de masas del eslabón  $i$  respecto al sistema de referencia del mundo  $M$ .

Aplicando entonces las expresiones 4.6 y 4.7 en 4.1, se tiene:

$$\mathfrak{L}(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{D}(\mathbf{q}) \cdot \dot{\mathbf{q}} - E_P(\mathbf{q}) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n d_{ij}(\mathbf{q}) \dot{q}_i \cdot \dot{q}_j - E_P(\mathbf{q}) \quad (4.9)$$

Ahora, para obtener las **ecuaciones del movimiento** es necesario efectuar previamente las siguientes derivadas:

$$\frac{\partial \mathfrak{L}}{\partial \dot{q}_k} = \sum_{j=1}^n d_{kj}(\mathbf{q}) \cdot \dot{q}_j \quad (4.10)$$

$$\begin{aligned} \frac{d}{dt} \left( \frac{\partial \mathfrak{L}}{\partial \dot{q}_k} \right) &= \sum_{j=1}^n d_{kj}(\mathbf{q}) \cdot \ddot{q}_j + \sum_{j=1}^n \frac{d}{dt} d_{kj}(\mathbf{q}) \cdot \dot{q}_j \\ \frac{d}{dt} \left( \frac{\partial \mathfrak{L}}{\partial \dot{q}_k} \right) &= \sum_{j=1}^n d_{kj}(\mathbf{q}) \cdot \ddot{q}_j + \sum_{i=1}^n \sum_{j=1}^n \frac{\partial d_{kj}}{\partial d_j} \cdot \dot{q}_i \cdot \dot{q}_j \end{aligned} \quad (4.11)$$

$$\frac{\partial \mathfrak{L}}{\partial q_k} = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \frac{\partial d_{ij}}{\partial q_k} \cdot \dot{q}_i \cdot \dot{q}_j - \frac{\partial E_P}{\partial q_k} \quad (4.12)$$

Reemplazando 4.10, 4.11 y 4.12 en 4.2, se tiene:

$$\sum_{j=i}^n d_{kj}(\mathbf{q}) \cdot \ddot{q}_j + \sum_{i=1}^n \sum_{j=1}^n \left( \frac{\partial d_{kj}}{\partial q_i} - \frac{1}{2} \frac{\partial d_{ij}}{\partial q_k} \right) \cdot \dot{q}_i \cdot \dot{q}_j - \frac{\partial E_P}{\partial q_k} = \tau_k \quad (4.13)$$

Esta expresión se puede reescribir como:

$$\sum_{j=i}^n d_{kj}(\mathbf{q}) \cdot \ddot{q}_j + \sum_{i=1}^n \sum_{j=1}^n c_{ijk}(\mathbf{q}) \cdot \dot{q}_i \cdot \dot{q}_j - \frac{\partial E_P}{\partial q_k} = \tau_k \quad (4.14)$$

Finalmente, se puede expresar la ecuación 4.14 en forma matricial para obtener la forma cerrada de la dinámica del sistema:

$$\mathbf{D}(\mathbf{q}) \cdot \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \cdot \dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} \quad (4.15)$$

donde  $\mathbf{D}(\mathbf{q})$  es la matriz inercial del sistema,  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$  es el término de Coriolis y  $\mathbf{g}(\mathbf{q})$  es el efecto de la gravedad.

Luego de este desarrollo matemático, se pueden aplicar las expresiones encontradas a la configuración particular del exoesqueleto, hallando las variables para cada eslabón que lo compone.

▪ Cálculo de la energía cinética:

Para el eslabón 1:

$$\mathbf{v}_{C1} = \begin{bmatrix} -l_{C1} \sin q_1 \\ l_{C1} \cos q_1 \end{bmatrix} \cdot \dot{q}_1 \quad (4.16)$$

$$\omega_1 = \dot{q}_1 \quad (4.17)$$

$$E_{C1} = \frac{1}{2} m_1 l_{C1}^2 \dot{q}_1^2 + \frac{1}{2} I_1^C \dot{q}_1^2 \quad (4.18)$$

Para el eslabón 2:

$$\mathbf{v}_{C1} = \begin{bmatrix} -l_1 \sin q_1 - l_{C1} \sin(q_1 + q_2) & -l_{C2} \sin(q_1 + q_2) \\ l_1 \cos q_1 + l_{C1} \cos(q_1 + q_2) & l_{C2} \cos(q_1 + q_2) \end{bmatrix} \cdot \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \quad (4.19)$$

$$\omega_2 = \dot{q}_1 + \dot{q}_2 \quad (4.20)$$

$$E_{C_2} = \frac{1}{2} m_2 [l_1^2 \dot{q}_1^2 + l_{C_2}^2 (\dot{q}_1 + \dot{q}_2)^2 + 2l_1 l_{C_2} (\dot{q}_1^2 + \dot{q}_1 \dot{q}_2) \cos q_2] + \frac{1}{2} I_{C_2}^C (\dot{q}_1 + \dot{q}_2)^2 \quad (4.21)$$

- Cálculo de la energía potencial:

Para el eslabón 1:

$$E_{P_1} = m_1 g l_{C_1} \sin q_1 \quad (4.22)$$

Para el eslabón 2:

$$E_{P_2} = m_2 g [l_1 \sin q_1 + l_{C_2} \sin(q_1 + q_2)] \quad (4.23)$$

- Cálculo del Lagrangiano:

$$\begin{aligned} \Omega(\mathbf{q}, \dot{\mathbf{q}}) &= E_{C_1} + E_{C_2} - E_{P_1} - E_{P_2} \\ &= \frac{1}{2} m_1 l_{C_1}^2 \dot{q}_1^2 + \frac{1}{2} I_{C_1}^C \dot{q}_1^2 \\ &\quad + \frac{1}{2} m_2 [l_1^2 \dot{q}_1^2 + l_{C_2}^2 (\dot{q}_1 + \dot{q}_2)^2 + 2l_1 l_{C_2} (\dot{q}_1^2 + \dot{q}_1 \dot{q}_2) \cos q_2] \\ &\quad + \frac{1}{2} I_{C_2}^C (\dot{q}_1 + \dot{q}_2)^2 - m_1 g l_{C_1} \sin q_1 - m_2 g [l_1 \sin q_1 + l_{C_2} \sin(q_1 + q_2)] \end{aligned} \quad (4.24)$$

Identificando términos con la ecuación 4.9 se tiene:

$$\begin{aligned} d_{11} &= m_1 l_{C_1}^2 + I_{C_1}^C + m_2 l_1^2 + m_2 l_{C_2}^2 + 2m_2 l_1 l_{C_2} \cos q_2 + I_{C_2}^C \\ d_{12} &= d_{21} = m_2 l_{C_2}^2 + m_2 l_1 l_{C_2} \cos q_2 + I_{C_2}^C \\ d_{22} &= m_2 l_{C_2}^2 + I_{C_2}^C \end{aligned} \quad (4.25)$$

- Cálculo de la ecuación del movimiento:

Teniendo en cuenta las ecuaciones 4.13 y 4.14 se tiene que:

$$c_{ijk} = \frac{\partial d_{kj}}{\partial q_i} - \frac{1}{2} \frac{\partial d_{ij}}{\partial q_k} \quad (4.26)$$

Por tanto, teniendo en cuenta las expresiones de 4.25:

$$\begin{aligned}
c_{111} &= \frac{\partial d_{11}}{\partial q_1} - \frac{1}{2} \frac{\partial d_{11}}{\partial q_1} = \frac{1}{2} \frac{\partial d_{11}}{\partial q_1} = 0 \\
c_{121} &= \frac{\partial d_{12}}{\partial q_1} - \frac{1}{2} \frac{\partial d_{12}}{\partial q_1} = \frac{1}{2} \frac{\partial d_{12}}{\partial q_1} = 0 \\
c_{112} &= \frac{\partial d_{21}}{\partial q_1} - \frac{1}{2} \frac{\partial d_{11}}{\partial q_2} = m_2 l_1 l_{C2} \sin q_2 \\
c_{122} &= \frac{\partial d_{22}}{\partial q_1} - \frac{1}{2} \frac{\partial d_{12}}{\partial q_2} = -\frac{1}{2} m_2 l_1 l_{C2} \sin q_2 \\
c_{211} &= \frac{\partial d_{11}}{\partial q_2} - \frac{1}{2} \frac{\partial d_{21}}{\partial q_1} = -2m_2 l_1 l_{C2} \sin q_2 \\
c_{221} &= \frac{\partial d_{12}}{\partial q_2} - \frac{1}{2} \frac{\partial d_{22}}{\partial q_1} = -m_2 l_1 l_{C2} \sin q_2 \\
c_{212} &= \frac{\partial d_{21}}{\partial q_1} - \frac{1}{2} \frac{\partial d_{21}}{\partial q_2} = \frac{1}{2} m_2 l_1 l_{C2} \sin q_2 \\
c_{222} &= \frac{\partial d_{22}}{\partial q_2} - \frac{1}{2} \frac{\partial d_{22}}{\partial q_2} = \frac{1}{2} \frac{\partial d_{22}}{\partial q_2} = 0
\end{aligned} \tag{4.27}$$

Haciendo:

$$\begin{aligned}
\phi_1 &= -\frac{\partial E_p}{\partial q_1} = m_1 g l_{C1} \cos q_1 + m_2 g l_1 \cos q_1 + m_2 g l_{C2} \cos(q_1 + q_2) \\
\phi_2 &= -\frac{\partial E_p}{\partial q_2} = m_2 g l_{C2} \cos(q_1 + q_2)
\end{aligned} \tag{4.28}$$

Por lo tanto, teniendo en cuenta la expresión 4.14, se tiene:

$$\begin{aligned}
\tau_1 &= \sum_{j=1}^n d_{1j} \cdot \ddot{q}_j + \sum_{i=1}^n \sum_{j=1}^n c_{ij1} \cdot \dot{q}_i \cdot \dot{q}_j + \phi_1 \\
&= d_{11} \cdot \ddot{q}_1 + d_{12} \cdot \ddot{q}_2 + c_{111} \cdot \dot{q}_1^2 + (c_{121} + c_{211}) \cdot \dot{q}_1 \cdot \dot{q}_2 + c_{221} \cdot \dot{q}_2^2 + \phi_1
\end{aligned} \tag{4.29}$$

$$\begin{aligned}\tau_2 &= \sum_{j=i}^n d_{2j} \cdot \ddot{q}_j + \sum_{i=1}^n \sum_{j=1}^n c_{ij2} \cdot \dot{q}_i \cdot \dot{q}_j + \phi_2 \\ &= d_{21} \cdot \ddot{q}_1 + d_{22} \cdot \ddot{q}_2 + c_{112} \cdot \dot{q}_1^2 + (c_{122} + c_{212}) \cdot \dot{q}_1 \cdot \dot{q}_2 + c_{222} \cdot \dot{q}_2^2 + \phi_2\end{aligned}\quad (4.30)$$

De esta manera se han encontrado los torques motores de cada articulación del exoesqueleto para el caso de tener dos grados de libertad, cadera y rodilla, pivotados sobre la cadera.

Con este desarrollo se puede apreciar que la metodología del Lagrangiano es bastante detallada en su modelo matemático y fácil de entender debido a los principios físicos que se aplican de balance de energía, pero la implementación de un algoritmo que pueda generar el modelo de un sistema con más grados de libertad resulta sumamente complejo e ineficiente para aplicaciones de prototipado y control en tiempo real, ya que implicaría un mayor retardo en la respuesta debido a la cantidad de operaciones involucradas que están relacionadas con los  $n$  grados de libertad del sistema en un factor de  $n^4$ .

#### 4.2.2 Formulación Newton-Euler

Esta formulación busca simplificar el algoritmo para describir la dinámica de un sistema de cuerpos rígidos y hacerlo viable de utilizar en aplicaciones de control en tiempo real. La formulación de Newton-Euler se basa en las ecuaciones que describen el movimiento lineal y angular.

$$\mathbf{F}_i = m_i \frac{d\mathbf{v}_{C_i}^M}{dt} = m_i \mathbf{a}_{C_i}^M \quad (4.31)$$

$$\boldsymbol{\tau}_i = \mathbf{I}_i^{C_i} \dot{\boldsymbol{\omega}}_i^{C_i} + \dot{\boldsymbol{\omega}}_i^{C_i} \times (\mathbf{I}_i^{C_i} \dot{\boldsymbol{\omega}}_i^{C_i}) \quad (4.32)$$

Como se observa, estas ecuaciones relacionan fuerzas y torques con las velocidades y aceleraciones del centro de masas de un eslabón, teniendo en cuenta su masa, longitud, posición e inercia.

Si se tiene en cuenta que en una cadena cinemática, cada eslabón está unido a otros por medio de las articulaciones y que en las ecuaciones de un eslabón aparecen fuerzas y pares acoplados que también aparecen en las ecuaciones del eslabón vecino, entonces se

pueden determinar los términos acoplados y obtener así una descripción dinámica del sistema. Para ello se emplea el principio de D'Alembert el cual adiciona la fuerza inercial y el momento por movimiento dinámico al balance estático de fuerzas y momentos.

Si se considera un sistema de referencia solidario al eslabón  $i$  situado en el centro de masas del mismo, además de los sistemas de referencia  $i-1$  e  $i$  de los eslabones  $i$  e  $i+1$  respectivamente, y un sistema de referencia estacionario M, se puede establecer lo siguiente:

Balance de fuerzas:

$$\begin{aligned} {}^{i-1}\mathbf{F}_i^M - {}^i\mathbf{F}_{i+1}^M + m_i \mathbf{g}_i &= m_i \mathbf{a}_{C_i}^M \\ {}^{i-1}\mathbf{F}_i^M &= m_i \mathbf{a}_{C_i}^M - m_i \mathbf{g}_i + {}^i\mathbf{F}_{i+1}^M \end{aligned} \quad (4.33)$$

Balance de momentos:

$$\begin{aligned} {}^{i-1}\boldsymbol{\tau}_i^{C_i} - {}^i\boldsymbol{\tau}_{i+1}^{C_i} + {}^{i-1}\mathbf{F}_i^M \times \mathbf{p}_{i-1,C_i}^i - {}^i\mathbf{F}_{i+1}^M \times \mathbf{p}_{i,C_i}^i &= I_i^{C_i} \dot{\boldsymbol{\omega}}_i^{C_i} + \dot{\boldsymbol{\omega}}_i^{C_i} \times (I_i^{C_i} \dot{\boldsymbol{\omega}}_i^{C_i}) \\ {}^{i-1}\boldsymbol{\tau}_i^{C_i} &= {}^i\boldsymbol{\tau}_{i+1}^{C_i} - {}^{i-1}\mathbf{F}_i^M \times \mathbf{p}_{i-1,C_i}^i + {}^i\mathbf{F}_{i+1}^M \times \mathbf{p}_{i,C_i}^i + I_i^{C_i} \dot{\boldsymbol{\omega}}_i^{C_i} + \dot{\boldsymbol{\omega}}_i^{C_i} \times (I_i^{C_i} \dot{\boldsymbol{\omega}}_i^{C_i}) \end{aligned} \quad (4.34)$$

Aplicando estas expresiones a nuestro sistema de análisis formado por un exoesqueleto de dos eslabones y un pivote, se tiene:

Proceso recursivo hacia adelante:

*Para el eslabón 1:*

$$\begin{aligned} \boldsymbol{\omega}_1 &= \dot{q}_1 \mathbf{z}_0 \\ \dot{\boldsymbol{\omega}}_1 &= \ddot{q}_1 \mathbf{z}_0 \end{aligned} \quad (4.35)$$

$$\begin{aligned} \mathbf{p}_{0,C_1}^1 &= l_{C_1} \mathbf{x}_1 \\ \mathbf{p}_{1,C_1}^1 &= (l_1 - l_{C_1}) \mathbf{x}_1 \end{aligned} \quad (4.36)$$

$$\mathbf{a}_{C_1}^0 = \ddot{q}_1 \mathbf{z}_0 \times \mathbf{p}_{0,C_1}^1 + \dot{q}_1 \mathbf{z}_0 \times (\dot{q}_1 \mathbf{z}_0 \times \mathbf{p}_{0,C_1}^1) = l_{C_1} \ddot{q}_1 \mathbf{y}_1 - l_{C_1} \dot{q}_1^2 \mathbf{x}_1 = \begin{bmatrix} -l_{C_1} \dot{q}_1^2 \\ l_{C_1} \ddot{q}_1 \end{bmatrix} \quad (4.37)$$

Para el eslabón 2:

$$\begin{aligned}\omega_1 &= (\dot{q}_1 + \dot{q}_2)z_1 \\ \dot{\omega}_1 &= (\ddot{q}_1 + \ddot{q}_2)z_1\end{aligned}\tag{4.38}$$

$$\begin{aligned}p_{1,C_2}^2 &= l_{C_2}x_2 \\ p_{2,C_2}^2 &= (l_2 - l_{C_2})x_2\end{aligned}\tag{4.39}$$

$$a_{C_2}^0 = \begin{bmatrix} -l_1\dot{q}_1^2 \cos q_2 + l_1\ddot{q}_1 \sin q_2 - l_{C_2}(\dot{q}_1 + \dot{q}_2)^2 \\ l_1\dot{q}_1^2 \sin q_2 + l_1\ddot{q}_1 \cos q_2 + l_{C_2}(\dot{q}_1 + \dot{q}_2)^2 \end{bmatrix}\tag{4.40}$$

Proceso recursivo hacia atrás:

*Balace de fuerzas en eslabón 2:*

$$\begin{aligned}{}^2F_3^0 &= 0 \\ {}^1F_2^0 &= m_2 a_{C_2}^0 - m_2 g_2\end{aligned}\tag{4.41}$$

donde

$$g_2 = g \begin{bmatrix} \sin(q_1 + q_2) \\ -\cos(q_1 + q_2) \end{bmatrix}\tag{4.42}$$

*Balace de pares para eslabón 2:*

$$\begin{aligned}{}^1\tau_2^{C_2} &= -{}^1F_2^0 \times p_{1,C_2}^2 + I_2^{C_2} \dot{\omega}_2^{C_2} + \omega_2^{C_2} \times (I_2^{C_2} \dot{\omega}_2^{C_2}) \\ {}^1\tau_2^{C_2} &= I_2^{C_2}(\ddot{q}_1 + \ddot{q}_2) + (m_2 l_1 l_{C_2} \sin q_2) \dot{q}_1^2 + (m_2 l_1 l_{C_2} \cos q_2) \ddot{q}_1 + m_2 l_{C_2}^2 (\ddot{q}_1 + \ddot{q}_2) \\ &\quad + m_2 g l_{C_2} \cos(q_1 + q_2)\end{aligned}\tag{4.43}$$

*Balace de fuerzas para el eslabón 1:*

$${}^0F_1^0 = m_1 a_{C_1}^0 - m_1 g_1\tag{4.44}$$

$$g_1 = g \begin{bmatrix} \sin q_1 \\ -\cos q_1 \end{bmatrix}\tag{4.45}$$

*Balace de pares para el eslabón 1:*

$${}^0\tau_1^{C_1} = {}^1\tau_2^{C_1} - {}^0F_1^0 \times p_{0,C_1}^1 + {}^1F_2^0 \times p_{1,C_1}^1 + I_1^{C_1} \dot{\omega}_1^{C_1} + \omega_1^{C_1} \times (I_1^{C_1} \dot{\omega}_1^{C_1})$$



$${}^0\tau_1^{C1} = {}^1\tau_2^{C2} + m_1 l_{C1}^2 \ddot{q}_1 + m_1 g l_{C1} \cos q_1 + m_2 g l_1 \cos q_1 + I_1^{C1} \ddot{q}_1 + m_2 l_1^2 \ddot{q}_1 - m_2 l_1 l_{C2} (\dot{q}_1 + \dot{q}_2)^2 \sin q_2 + m_2 l_1 l_{C2} (\ddot{q}_1 + \ddot{q}_2) \cos q_2 \quad (4.46)$$

De esta manera se obtiene el mismo resultado que la formulación de Lagrange pero con una disminución importante en el número de operaciones necesarias para encontrar la respuesta cerrada, lo cual hace que este método sea más versátil para su implementación computacional.

### 4.2.3 Modelamiento gráfico con Matlab/Simulink y SimMechanics

Luego de haber desarrollado el modelamiento de una sola pierna en fase de balanceo con la formulación de Lagrange y con la formulación de Newton-Euler, se ha podido observar que la descripción de un sistema mecánico dinámico, en especial la marcha humana, es un tema matemáticamente complejo, por tanto la eficiencia y el coste computacional del sistema que se desarrolle depende de las técnicas de solución y los modelos que se adopten.

De esta manera, se ha recurrido a evaluar un tercer método de modelamiento del exoesqueleto de pierna, basado en el empleo de herramientas computacionales de simulación de sistemas mecánicos, específicamente las ofrecidas por la compañía Mathworks con su software *Matlab/Simulink* y su librería *SimMechanics*, pues pese a que el proceso de modelado matemático es más detallado, y en el caso analizado, relativamente sencillo para un mecanismo con dos grados de libertad, estos planteamientos incrementan su complejidad en proporción a los grados de libertad que se consideren, mientras que la plataforma de simulación mantiene su versatilidad para el modelado y nos permite realizar validaciones de manera eficaz y rápida.

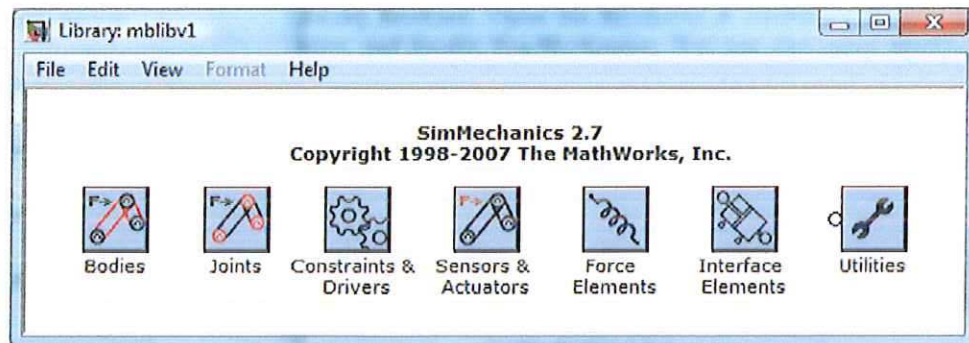
#### 4.2.3.1 Descripción general de SimMechanics

El software SimMechanics es un entorno de modelamiento basado en diagramas de bloques para el diseño, análisis y simulación de sistemas de cuerpos rígidos, empleando para ello la dinámica Newtoniana estándar de fuerzas y torques. Con esta herramienta computacional se pueden modelar y simular sistemas mecánicos complejos a través de una serie de bloques que permiten definir cuerpos y juntas, así como especificar las propiedades de éstos, como la masa, el tensor de inercia, restricciones cinemáticas,

trayectoria de movimientos, sistemas de coordenadas, centros de masa, condiciones iniciales, entre otras. La herramienta permite a su vez incluir sensores y actuadores en el sistema para visualizar por medio de gráficas de curvas y gráficos 3D, los movimientos y respuestas del sistema.

A continuación se describe brevemente el entorno de SimMechanics para luego desarrollar el modelamiento del mismo sistema (dos eslabones y dos articulaciones rotacionales) analizado con las formulaciones matemáticas de solución cerrada, con el fin de contrastar las diferentes metodologías empleadas en el proceso de modelado y simulación.

Figura 20. Librerías de SimMechanics dentro del entorno Simulink de Matlab



Fuente: The Mathworks, Inc. *SimMechanics 2.7*

En la figura 20 se pueden observar las librerías de la herramienta SimMechanics, las cuales están clasificadas de acuerdo al tipo de bloque mecánico. Cada uno de estos bloques se desglosa a su vez en otras sublibrerías. A continuación se define el propósito de estas librerías:

- **Cuerpo (Body):** esta librería contiene los bloques necesarios para que el usuario defina los cuerpos rígidos de su modelo, indicando su respectiva masa, inercia, posición y orientación, y los sistemas de coordenadas locales (CSs) asociados al cuerpo. La librería también presenta el bloque *Ground* que representa un punto fijo en el espacio sobre el cual se referencian los cuerpos, articulaciones y movimientos; y el bloque *Machine Environment* que permite especificar las opciones del entorno mecánico del diagrama de bloques.
- **Articulaciones (Joints):** esta librería provee los bloques para representar el movimiento relativo entre los cuerpos como grados de libertad. Se presenta una gran cantidad de sublibrerías de acuerdo al tipo de movimiento asociado a ese

grado de libertad. Básicamente los bloques que presenta se derivan de las articulaciones rotacionales y prismáticas, y cada uno de ellos permite configurar los ejes sobre los que funciona dicha articulación.

- *Restrictores y manejadores (Constraints & Drivers)*: esta librería presenta los bloques necesarios para establecer restricciones y controles en los grados de libertad del sistema, por ejemplo limitaciones de posición, velocidad, fuerza, etc.
- *Sensores y actuadores (Sensors & Actuators)*: permiten monitorear las variables dinámicas de los cuerpos y articulaciones, establecer sus condiciones iniciales o aplicar fuerzas, torques o trayectorias de movimiento directamente.
- *Elementos de fuerza (Force elements)*: estos bloques permiten modelar las fuerzas internas del sistema.
- *Elementos de interfaz (Interface elements)*: esta librería permite conectar sistemas 3d modelados en SimMechanics con sistemas unidimensionales de otras aplicaciones de Simulink.
- *Utilidades (Utilities)*: es una librería que contiene bloques misceláneos que son de utilidad para el modelado, así como también permite contener las librerías desarrolladas por el usuario.

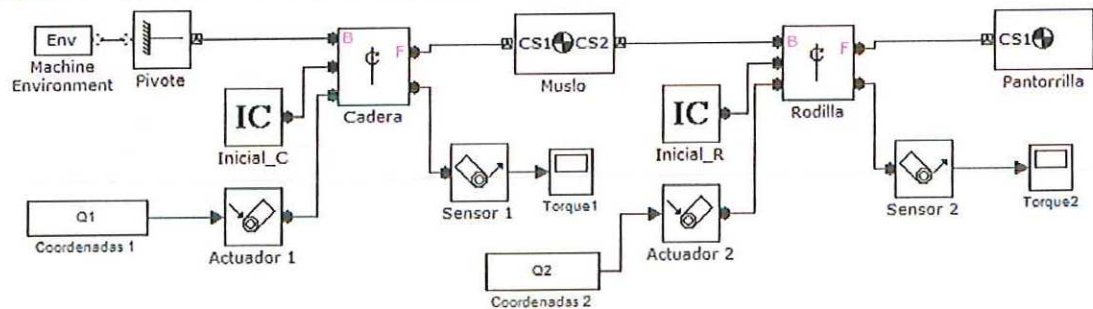
#### 4.2.3.2 Construcción del modelo de una pierna en SimMechanics

Para modelar el sistema del exoesqueleto analizado en las formulaciones matemáticas, se realizaron los siguientes pasos en la hoja de trabajo de Simulink:

- a. Selección, posición e interconexión de los bloques que definen la cadena cinemática de interés, incluyendo el sistema inercial de referencia, los cuerpos rígidos y las articulaciones rotacionales.
- b. Especificación de las propiedades de cada bloque, como sistemas coordenados locales, centros de gravedad, masas, tensores de inercia, ejes de acción, dimensiones.
- c. Conexión de los sensores y actuadores para definir las visualizaciones de las variables de interés y asignar las fuerzas, torques y condiciones de movimiento deseadas.
- d. Solución y visualización del movimiento del sistema para las condiciones dadas.

En la figura 21 se presenta el modelo de dos grados de libertad de la pierna pivotada en la cadera. Se puede observar la facilidad de la herramienta computacional para establecer el sistema de cuerpos rígidos unidos por articulaciones rotacionales. Las entradas del sistema son los bloques que contienen el vector de coordenadas generalizadas de posición, velocidad y aceleración para cada eslabón. La salida del sistema son los sensores que visualizan el torque calculado por dinámica inversa para dichos movimientos.

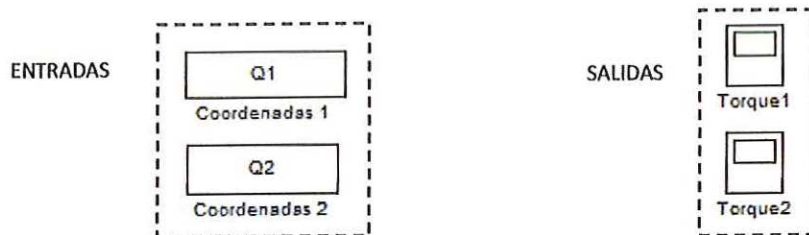
**Figura 21.** Modelado del sistema muslo-pantorrilla en SimMechanics



Fuente: El autor del proyecto

Como se puede observar, la estrategia de modelado con esta herramienta es bastante sencilla, sólo es necesario seguir un procedimiento básico para la definición de los elementos, sus parámetros y relaciones, tal como se hace matemáticamente, sólo que en este caso las definiciones y asignaciones de propiedades se hacen interactuando con la interfaz de cada bloque gráfico. En las siguientes figuras (22-30) se detallan las especificaciones necesarias para definir cada tipo de bloque empleado en el modelo anterior.

**Figura 22.** Bloques de entrada y salida

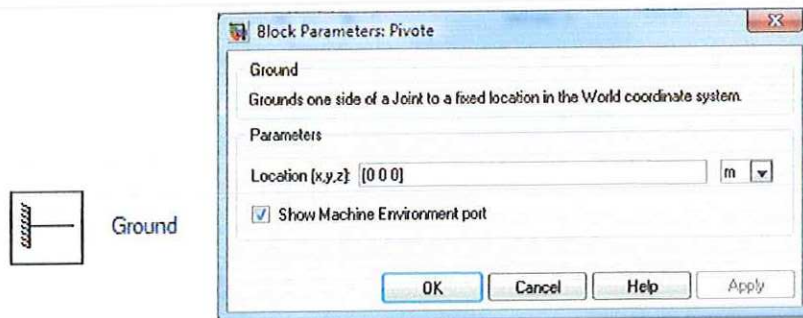


Fuente: El autor del proyecto

En la figura 22 se observan los bloques de entradas y salidas del sistema, específicamente las entradas son Q1 y Q2 que son las matrices de coordenadas articulares generalizadas, las cuales están formadas por 4 vectores columnas. La primera columna es el tiempo, la

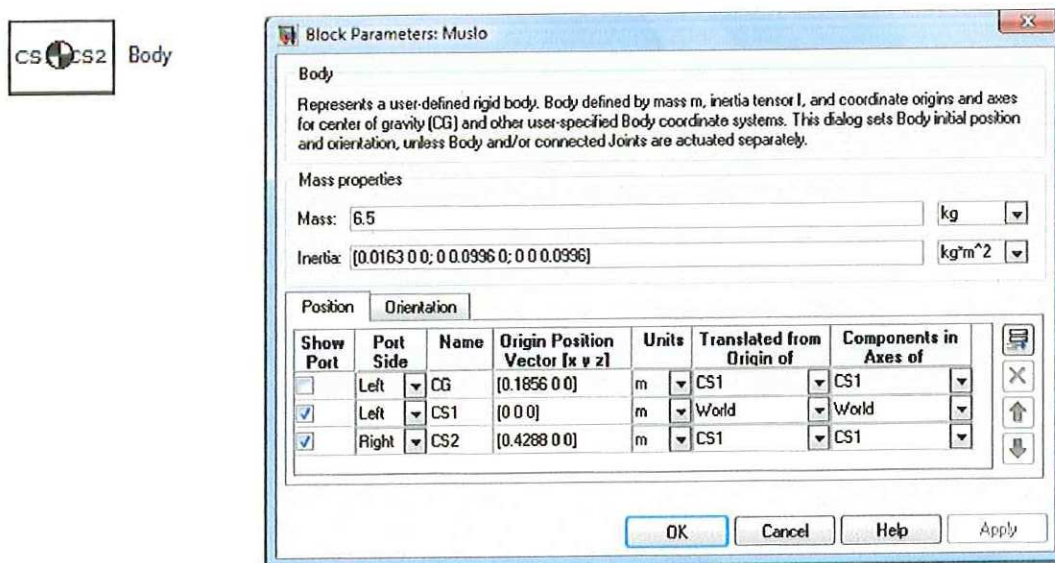
segunda columna es el ángulo de la articulación respectiva para cada instante de tiempo, la tercera columna es la velocidad angular del eslabón móvil de esa articulación y la cuarta columna es la aceleración angular del mismo. En este caso, Q1 se refiere a la articulación de la cadera y Q2 a la articulación de la rodilla. Por otro lado las salidas están representadas por los bloques Torque 1 y Torque 2, los cuales son simplemente visualizaciones gráficas de los momentos articulares en función del tiempo.

Figura 23. Propiedades del bloque *Ground*



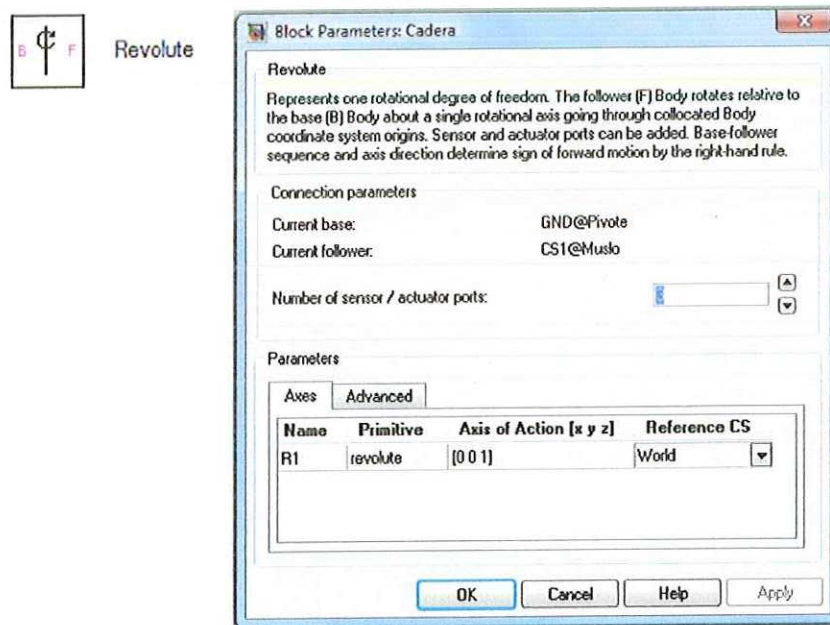
En la figura 23 se puede apreciar la definición del sistema inercial global del modelo, cuyo centro se localiza espacialmente en el punto  $[x,y,z]=[0\ 0\ 0]$ . También se define la unidad de longitud del sistema en metros [m].

Figura 24. Propiedades del bloque *Body*



En la figura 24 se puede apreciar cómo se definen los cuerpos rígidos que forman el sistema. Los datos empleados para el tensor de inercia fueron obtenidos aproximando las partes de la pierna a formas cilíndricas cuya masa y longitud son función de la masa y de la altura total de una persona, tomando como base las relaciones establecidas en las tablas antropométricas de Winter. Cabe destacar a su vez que el tensor de inercia está referenciado al centro de masas del cuerpo, a fin de que el producto de los momentos de inercia secundarios se anule y sea necesario definir sólo la diagonal de los momentos inerciales principales para la triada del sistema de coordenadas del cuerpo. Por otro lado se observa en las propiedades del bloque que se establecen posiciones de sistemas de referencia locales asociados a cada eslabón, los cuales definen el orden de las relaciones con los eslabones vecinos. Indirectamente, con dichas posiciones se están asignando las longitudes y orientaciones de los eslabones. Es importante mantener concordancia con las unidades empleadas en las especificaciones de cada bloque.

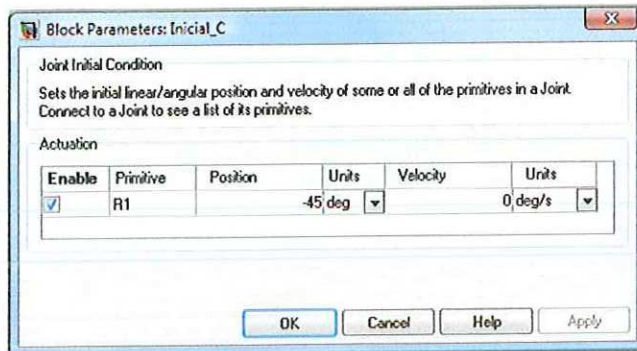
Figura 25. Propiedades del bloque *Revolute*



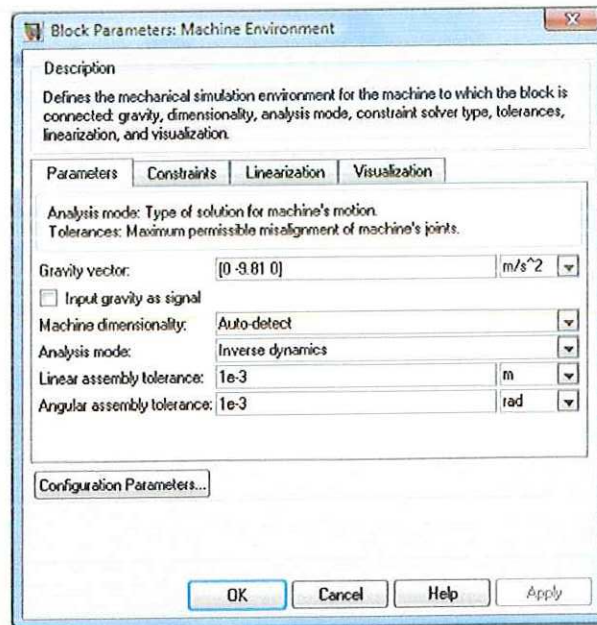
En la figura 25 se aprecian las propiedades de una articulación rotacional, en la cual solo es necesario definir la cantidad de sensores y actuadores asociados a la junta, el eje de rotación y el sistema coordinado de referencia. También se presenta la información de los dos elementos que une dicha articulación, definiendo una base y un seguidor.

Figura 26. Propiedades del bloque *Joint Initial Condition* y *Machine Environment*

IC Joint Initial Condition

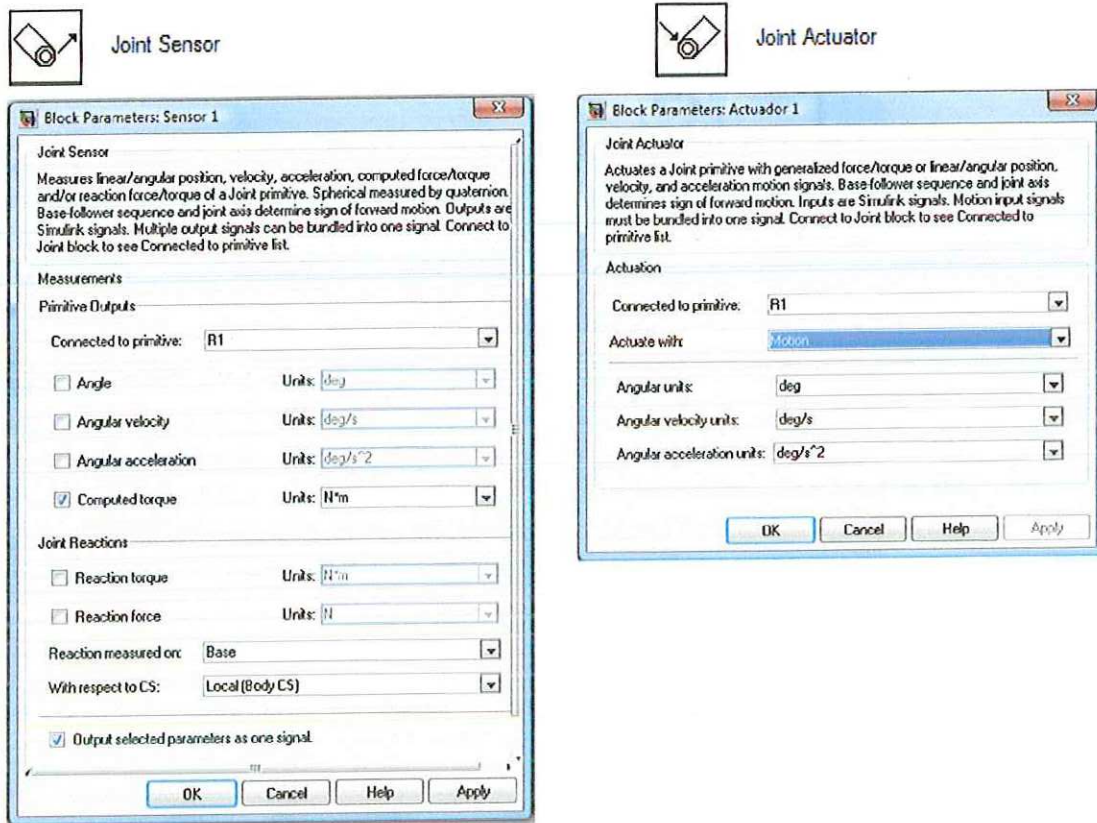


Env Machine Environment



En la figura 26 se muestra la definición de las condiciones iniciales de una articulación, así como las propiedades del bloque de máquina de entorno, en el cual se definen las opciones de operación y solución del diagrama completo. Se puede apreciar que se define el vector de gravedad y el modo de análisis; en este caso ha sido seleccionado el modo de dinámica inversa pues se parte de una trayectoria deseada de los cuerpos para hallar los torques necesarios que logren esa respuesta en el tiempo.

Figura 27. Propiedades de los bloques *Sensor* y *Actuator*

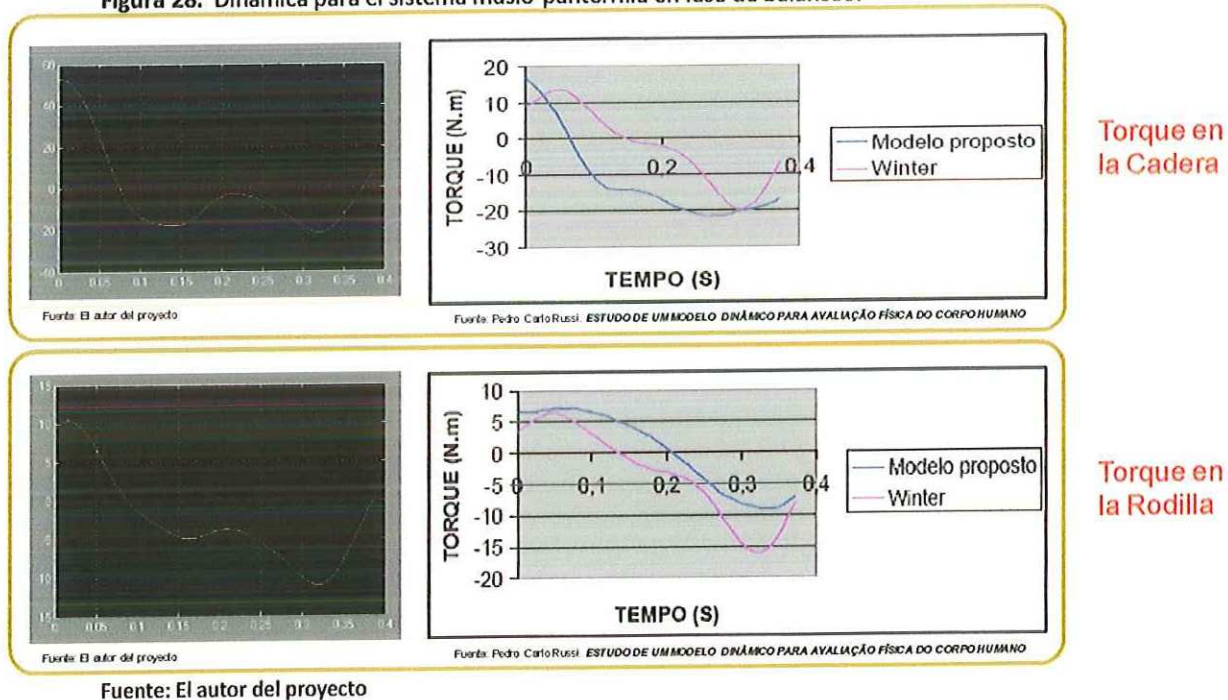


En la figura 27 se pueden observar las definiciones y opciones que presentan los bloques de sensores y actuadores. El bloque del sensor permite obtener como salida el ángulo de la articulación, la velocidad angular, la aceleración angular, y el torque computado que es el parámetro de interés en este trabajo. También se observa que ese mismo bloque de sensor permite obtener las fuerzas y torques de reacción que se generan en la articulación. Por otro lado el bloque del actuador permite dos modos de funcionamiento, uno en el cual la salida del actuador son las fuerzas generalizadas, empleadas para la dinámica directa del sistema, y otro modo que describe el movimiento deseado en la articulación a través de las coordenadas generalizadas de posición, velocidad y aceleración angular, que serían las variables de entrada para evaluar los formalismos de modelado desarrollados.



En la figura 28 se pueden contrastar los resultados de la dinámica para una trayectoria de balanceo de la pierna, para lo cual se emplearon los valores de las variables articulares de posición, velocidad y aceleración angular, reportados en un completo análisis de la marcha humana realizado por D. Winter empleando técnicas de captura de movimiento por video. Las curvas presentadas a la derecha de la figura, fueron resultados recopilados de la investigación de C. Russi quien propuso un modelo dinámico simplificado para la marcha humana basado en la formulación Lagrangiana pero despreciando el tensor de inercia de los cuerpos y por ende eliminando el factor de energía cinética angular descrito en el apartado 4.2.1. La curva presentada en la parte izquierda de la figura representa la variación del torque en el tiempo según el modelo computacional desarrollado por el autor del proyecto.

Figura 28. Dinámica para el sistema muslo-pantorrilla en fase de balanceo.



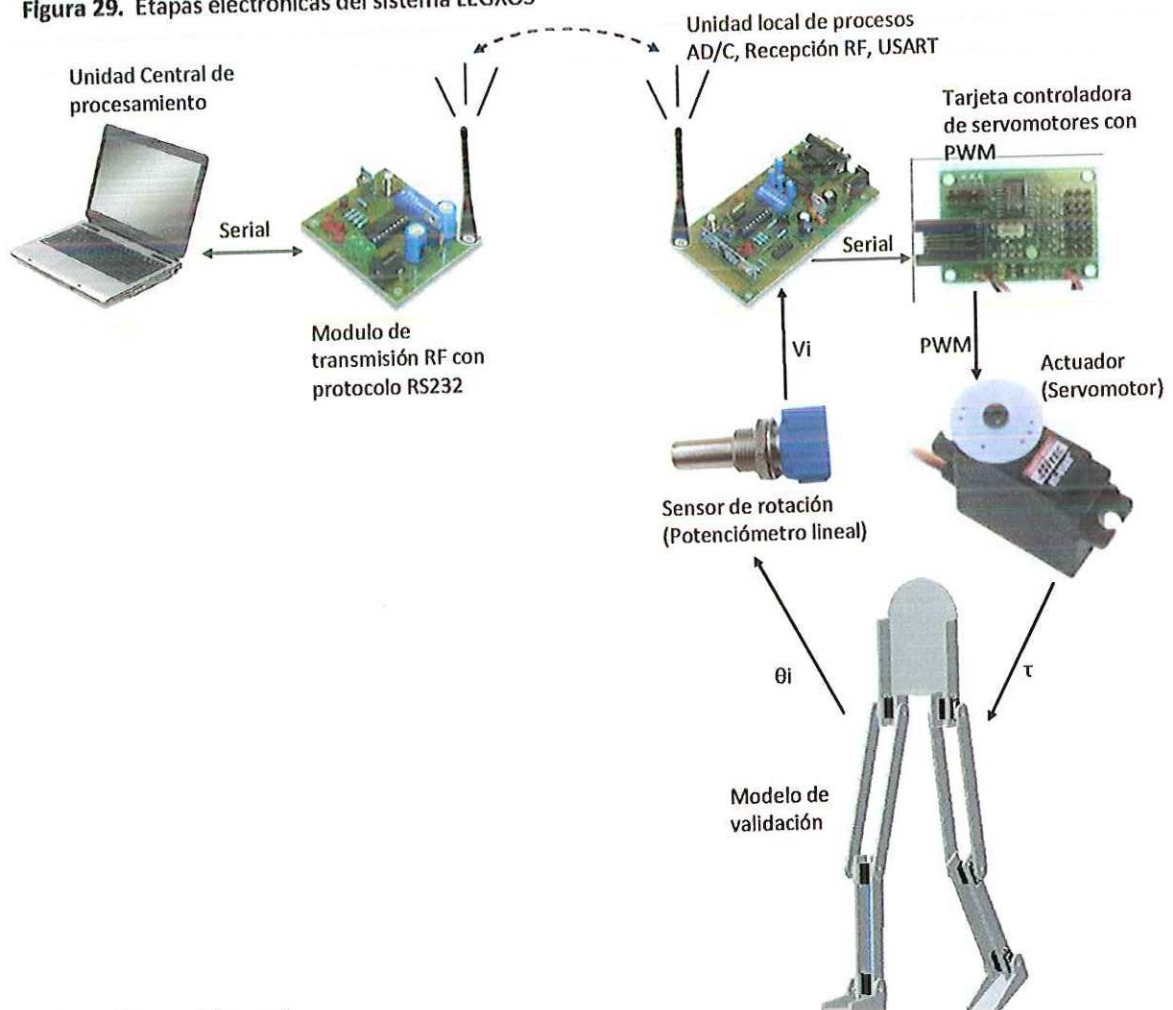
Fuente: El autor del proyecto

Se puede apreciar como la curva en amarillo sigue en su contorno a la curva en color púrpura del análisis de Winter, mientras que el modelo propuesto por Russi se aleja bastante del comportamiento dinámico real, lo cual es de esperarse debido a la omisión de la energía cinética angular que se realiza en dicha formulación. De esta manera se puede validar la herramienta SimMechanics para desarrollar el modelo completo de la marcha, que será descrito en el apartado de simulaciones del capítulo 7.

## 5. DISEÑO ELECTRÓNICO

En este capítulo se presenta el detalle de la instrumentación electrónica involucrada en el desarrollo del sistema, que permite convertir las señales de desplazamiento angular, en una magnitud eléctrica, para luego transmitirla y procesarla a fin de obtener visualizaciones y respuestas del sistema para accionar y controlar los actuadores. El diseño electrónico es de gran importancia ya que nos permite establecer relaciones entre la mente y el cuerpo del sistema, de la manera más eficaz posible. En la figura 29 se presenta un diagrama con las etapas electrónicas desarrolladas en este proyecto.

Figura 29. Etapas electrónicas del sistema LEGXOS



Fuente: El autor del proyecto.

## 5.1 SENSORES

La unidad de medición del sistema está basada en sensores de desplazamiento angular, específicamente se emplearon potenciómetros lineales de tipo rotacional, ya que cumplen con los requerimientos necesarios para un bioinstrumento de medida, que son: no altera el evento que está midiendo, presenta linealidad en la amplitud, respuesta en frecuencia adecuada y ausencia de distorsión de fase; además de su sencillez de funcionamiento y robustez física para acoples mecánicos.

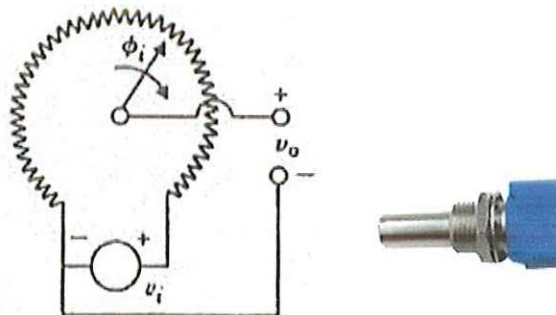
El funcionamiento del potenciómetro se basa en la variación de su resistencia de acuerdo a la posición  $\theta$  de su eje. Es un instrumento de orden cero, ya que no sigue un comportamiento de acuerdo a ninguna ecuación diferencial. Su respuesta viene dada por:

$$R = (1 - \beta) \cdot R_t \quad (5.1)$$

donde  $R_t$  es la resistencia total,  $\beta$  es un factor que depende de la posición angular .

El potenciómetro es un elemento usado comúnmente debido a su facilidad para entregar una señal eléctrica en tiempo real de la magnitud física que mide. Se le clasifica como sensor activo porque requiere una fuente de voltaje externa para poder entregar su salida eléctrica. En la figura 30 se observa este principio de funcionamiento:

Figura 30. Potenciómetro rotacional



Ya que el sistema LEGXOS está basado en manejo de información a través de señales eléctricas, es necesario entonces obtener la relación eléctrica del potenciómetro. Por la definición de un divisor de tensión y la variación resistiva del potenciómetro, se tiene que:

$$V_o = \frac{V_i R}{R_t} = \frac{V_i(1 - \beta) \cdot R_t}{R_t} = V_i(1 - \beta) \quad (5.2)$$

Como la resistencia del potenciómetro está en función de su posición de una manera lineal, de igual manera lo estará su voltaje de salida, por ello se expresa la ecuación de interés que caracteriza nuestro sistema de medida en función de la magnitud física que mide:

$$V_o = V_i \left( \frac{\theta}{\theta_{max}} \right) \quad (5.3)$$

### 5.1.1 Caracterización del sensor utilizado

A pesar de que los potenciómetros lineales aseguran su respuesta lineal, es necesario realizar la caracterización experimental del elemento, a fin de tener la certeza y la precisión real del rango y respuesta del sensor.

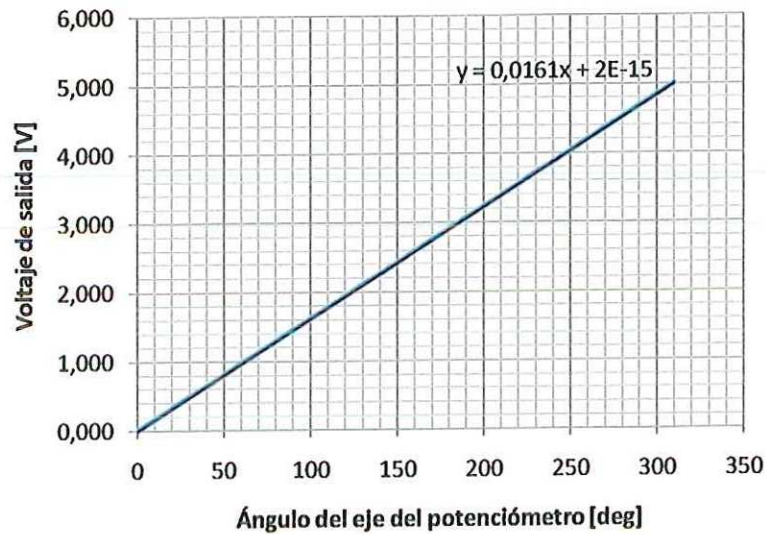
En este proyecto se utilizaron potenciómetros rotacionales, con barrido de 310°, de una vuelta y resistencia máxima de 5K ohm, marca *Bourns*, con número de referencia 51AADC20A13. A continuación se muestran los datos tabulados con la variación del voltaje de salida en función del ángulo del eje del potenciómetro, su gráfica y su ecuación experimental.

**Tabla 2.** Datos experimentales de la respuesta del sensor.

Ángulo [deg]	Resistencia [kohm]	Voltaje [V]
0	0,000	0,000
10	0,168	0,161
20	0,337	0,323
30	0,505	0,484
40	0,674	0,645
50	0,842	0,806
60	1,010	0,968
70	1,179	1,129
80	1,347	1,290
90	1,515	1,452
100	1,684	1,613
110	1,852	1,774
120	2,021	1,935
130	2,189	2,097
140	2,357	2,258
150	2,526	2,419
160	2,694	2,581
170	2,863	2,742
180	3,031	2,903
190	3,199	3,065
200	3,368	3,226
210	3,536	3,387
220	3,705	3,548
230	3,873	3,710
240	4,041	3,871
250	4,210	4,032
260	4,378	4,194
270	4,546	4,355
280	4,715	4,516
290	4,883	4,677
300	5,052	4,839
310	5,220	5,000

Fuente: El autor del proyecto

**Figura 31.** Respuesta experimental del sensor de posición angular empleado



Fuente: El autor del proyecto

En la gráfica anterior se confirma la respuesta lineal del potenciómetro seleccionado en función del ángulo de su eje. El ángulo instantáneo crece hacia el ángulo de barrido máximo con una pendiente de 0.0161, que viene dada por el voltaje de alimentación del sensor (5V) y el ángulo mecánico máximo de su eje (310°).

## 5.2 UNIDAD DE PROCESAMIENTO LOCAL

Como se describe en el capítulo 3, esta unidad consta de una etapa de adquisición y conversión análogo-digital, y de una etapa controladora de actuadores. Las operaciones de la unidad están implementadas a través de microcontroladores, uno para cada etapa.

La operación de esta unidad está comandada por instrucciones enviadas inalámbricamente desde la unidad central de procesamiento (computador). Se define como local para minimizar los efectos de ruido eléctrico y pérdida de datos presentes en transmisión de señales. Es recomendable que la tarjeta de adquisición esté lo más cercano posible a los sensores que generan las señales analógicas de interés. Por otro lado, al ser una unidad remota brinda mayor comodidad al sujeto que utiliza el exoesqueleto. A continuación se describen los módulos de operación implementados.

### 5.2.1 Comunicación Serial RS-232

La unidad local de procesamiento se comunica con la unidad central por medio del estándar de comunicación serial RS-232, el cual define la interfaz mecánica, los pines, las señales eléctricas y los protocolos que debe cumplir la comunicación serial. En el sistema LEGXOS se implementa una comunicación serial en modo asíncrono, para permitir el flujo bidireccional simultáneo de datos entre ambos dispositivos.

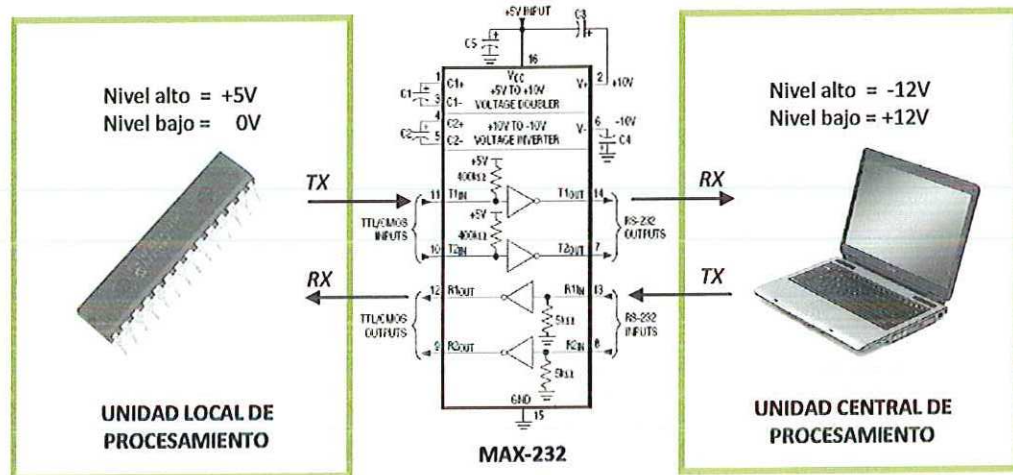
Entre las principales características del protocolo de comunicación se establecieron los siguientes valores;

- *Velocidad de transmisión: 2400.*  
Indica el número de bits por segundo que se transfieren, y se mide en baudios.
- *Bits de datos: 8.*  
Se refiere a la cantidad de bits en la transmisión. Cuando la computadora envía un paquete de información, el tamaño de ese paquete no necesariamente será de 8 bits. Un paquete se refiere a una transferencia de byte, incluyendo los bits de inicio/parada, bits de datos, y paridad.
- *Paridad: Ninguna.*  
Es una forma sencilla de verificar si hay errores en la transmisión serial. Existen los siguientes tipos de paridad: par, impar, marcada, espaciada, ninguna.

Respecto a las señales eléctricas, la norma RS-232 establece que un "1" lógico es un voltaje comprendido entre -5V y -15V en el transmisor y entre -3V y -25V en el receptor; y un "0" lógico es un voltaje comprendido entre +5V y +15V en el transmisor y entre +3V y +25V en el receptor. Por ello, el envío de niveles lógicos (bits) a través de cables o líneas de transmisión necesita la conversión a voltajes apropiados. En los microcontroladores para representar un 0 lógico se trabaja con voltajes inferiores a 0.8V, y para un 1 lógico con voltajes mayores a 2.0V. En general cuando se trabaja con familias TTL y CMOS se asume que un "0" lógico es igual a 0V y un "1" lógico es igual a 5V.

De esta manera, se implementa entonces un circuito que soluciona los problemas de niveles de voltaje cuando se requiere enviar unas señales digitales sobre una línea RS-232. Consiste en un circuito integrado que se utiliza en aquellas aplicaciones donde no se dispone de fuentes dobles de +12 y -12 Volts, llamado MAX-232, el cual necesita solamente una fuente de +5V para su operación, internamente tiene un elevador de voltaje que convierte el voltaje de +5V al de doble polaridad de +12V y -12V. En la figura 32 se presenta el esquema general de comunicación.

Figura 32. Diagrama de comunicación serial entre pc y microcontrolador



Fuente: El autor del proyecto

Como se mencionó anteriormente, la unidad local de procesamiento está formada por dos etapas diferentes: la tarjeta de adquisición y conversión análogo-digital, y la tarjeta controladora de servomotores. Ambas tarjetas reciben por el mismo medio las instrucciones desde la unidad central de procesamiento, por ello al establecer la comunicación, entre el sistema maestro y el esclavo, es necesario que exista una diferenciación de los comandos que operan cada tarjeta de la unidad local.

De esta manera se establecieron las siguientes pautas en el intercambio de datos:

*En la unidad central de control:*

- Orden de muestreo: PC envía el carácter 's' que es el identificador para la tarjeta de adquisición análogo digital.
- Recepción de muestras: Si existen datos en buffer serial, se lee el primer byte disponible; si éste es igual al carácter 's', se realizan 3 lecturas consecutivas y guarda cada byte en un vector diferente para cada señal muestreada.
- Orden de accionamiento de motores: PC envía en binarios el vector [255 motor posición], donde el primer número equivale al identificador para la tarjeta de servomotores, el segundo elemento (motor) indica a dicha tarjeta el actuador que desea mover, y el tercer elemento (posición) asigna la posición angular deseada para el eje del motor.



*En la unidad local:*

- Si el dato recibido es el carácter 's', entra en funcionamiento el Microcontrolador 1 de la tarjeta de adquisición y conversión análogo digital, quien devuelve el carácter 's' recibido y tres bytes más, uno para el resultado de la conversión de cada señal de posición angular muestreada (cadera, rodilla, tobillo).
- Si el dato recibido es el decimal '255', entonces entra en acción el Microcontrolador 2 de la tarjeta de servomotores. Esta lee luego dos bytes seguidos, el primero para seleccionar el motor y el segundo para indicar su posición.

Es importante aclarar que cada tarjeta de la unidad local de procesamiento se diseñó independiente, con el fin de que el sistema fuera modular y brindar así una mayor flexibilidad al uso de los circuitos electrónicos diseñados, puesto que tanto las tarjetas de adquisición como las tarjetas para control de servomotores, son empleadas en gran número de sistemas mecatrónicos. Los esquemáticos, hojas de datos y detalles del diseño electrónico se pueden encontrar en el anexo B.

#### **5.2.1.1 Interfaz de comunicación inalámbrica por RF**

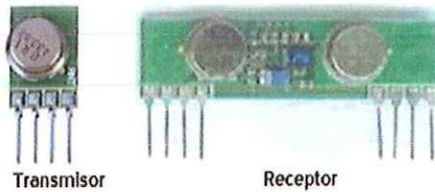
Para dar una mayor libertad al modelo, se planteó el uso de comunicación inalámbrica como un valor agregado al sistema. Para ello se implementaron módulos de radio frecuencia. Dentro de las referencias a las cuales se tuvo acceso para la selección final se encuentran:

Característica	TXM-433-LR RXM-433-LR	TXE-433-KH RXD-433-KH	TWS-434 A RWS-434
Tipo de montaje	Superficial	Superficial	DIP
Velocidad máxima	10 Kbps	10 Kbps	3 Kbps
Alcance máximo	3000 pies	300 pies	400 pies
Frecuencia	433 MHz	433 MHz	433 MHz
Costo	\$ 86.000	\$ 104.500	\$ 68.000

Teniendo en cuenta que la aplicación de este proyecto está orientada a centros de rehabilitación, se estipula una distancia máxima de 10m de posible separación entre la unidad central de procesos y el paciente. De esta manera, se ha seleccionado la referencia del transmisor y receptor: TWS-434A y RWS-434 de Reynolds Electronics (ver figura 33), por su relación costo-beneficio, ya que a pesar de ser los más económicos y de tener menor velocidad de transmisión de datos que los otros módulos, éstos tienen un alcance considerable de 85m en campo abierto y de aproximadamente 42m en espacios cerrados con paredes, lo cual satisface con gran margen de seguridad el requerimiento de distancia planteado.

Por otro lado, los dispositivos seleccionados presentan un patillaje común a los demás circuitos integrados que se emplean en el diseño electrónico que no son de tipo superficial, lo cual facilita la fabricación de la tarjeta electrónica. Las especificaciones de estos módulos son detalladas en el anexo C.

**Figura 33.** Módulos de Radiofrecuencia empleados



Fuente: El autor del proyecto

En la comunicación asíncrona, cuando no se realiza ningún intercambio de datos, la línea del transmisor es pasiva y se encuentra en un estado alto. Para empezar a transmitir un dato, el transmisor coloca en bajo esta línea durante el tiempo de un bit, lo que se conoce como bit de arranque, luego se inicia la transmisión del dato con el mismo intervalo de los bits correspondientes a la velocidad de envío seleccionada.

La transmisión se inicia con el LSB y se termina con el MSB, luego si se desea se agrega al terminar un bit de paridad y se incluyen uno o dos bits de parada. En este punto, la línea se encuentra nuevamente en estado alto.

Para la operación del sistema presentado en este proyecto se ha seleccionado una velocidad de transmisión de 2400 bits por segundo, sin paridad y con un bit de parada. Esta velocidad fue seleccionada debido a los límites de funcionamiento referenciados por los fabricantes de los módulos empleados, ya que a velocidades de transmisión superiores ocurre inestabilidad y pérdida de datos que afectan el desempeño del sistema.

#### Transmisor:

El modulo transmisor TWS-434 recibe los datos digitales que envía y los modula en amplitud, luego los transmite por radio a una frecuencia de 433MHz, para lo cual se emplea una antena de radio corriente de 42 cm de longitud.

#### Receptor:

El modulo receptor de radiofrecuencia ubicado en la unidad local del sistema esclavo, recibe los datos por la antena y los entrega tanto a la tarjeta de conversión análogo digital como a la tarjeta controladora de servomotores, para que operen de acuerdo a las pautas especificadas para la comunicación serial en el apartado 5.2.2.

Es importante señalar en este punto que el receptor presenta dos opciones para la entrega de los datos recibidos. Una patilla proporciona la información de manera digital con niveles lógicos de 0V a 5V, y otra patilla llamada salida lineal, arroja los datos recibidos de manera analógica, variando desde 2.3V para un nivel bajo a 2.5V para un nivel alto.

Al experimentar con ambas salidas se pudo encontrar la diferencia en sus respuestas. La salida digital sostiene sus valores de acuerdo a los flancos de subida y bajada de cada bit, de tal manera que si se envía un 1 lógico y se sostiene sin cambio de estado, la salida digital del receptor se activa sólo por el tiempo característico de la comunicación serial, en cambio la salida lineal sigue de forma continua la señal transmitida.

Con esto presente, se decidió trabajar con la salida lineal del receptor, pues su estabilidad ante la distancia resultó ser mayor. Para digitalizar esta salida lineal antes de enviar el dato a los microcontroladores, se empleó un circuito comparador LM311, con un umbral de 2.4V para la separación de niveles altos y bajos. Este nivel de tensión de comparación se determinó experimentalmente buscando la mejor respuesta de los módulos, para ello se realizó una prueba en la cual se enviaba una secuencia de datos conocida por el transmisor y se comparaba con la secuencia de datos del receptor.

## 5.2.2 Conversión Análogo-Digital

Las señales analógicas tienen la información en su amplitud o duración, de modo que para procesarlas con circuitos digitales, primero hay que adaptarlas y digitalizarlas, para lo cual se emplea un *convertidor análogo-digital (CA/D)*.

La entrada del convertidor A/D es una señal analógica y la salida es una palabra binaria que representa el nivel de la señal de entrada. Existen diferentes tipos de convertidores A/D, algunos vienen en circuitos integrados especializados para ello y otros vienen integrados en otros dispositivos. En este proyecto se emplea el convertidor interno del microcontrolador PIC16F873 de la gama media de Microchip, el cual funciona por el método de aproximaciones sucesivas.

En la conversión A/D es importante tener presente algunos parámetros que determinan el correcto funcionamiento del convertidor, como son: la frecuencia de la señal, la frecuencia de muestreo, el número de bits de la palabra digital, el rango dinámico y el tiempo de digitalización.

El convertidor A/D empleado presenta un tiempo de adquisición de 20 microsegundos aproximadamente y un tiempo de conversión de 23 microsegundos, por lo tanto el período de digitalización de cada señal articular es de 43 microsegundos. Para lograr una correcta adquisición de la cinemática articular de la pierna, se seleccionó una frecuencia de muestreo de 1000 muestras por segundo (1KHz), que cumple con el teorema del muestreo de Nyquist-Shannon, en el que se enuncia que la frecuencia de muestreo debe ser mayor al doble de la frecuencia máxima del sistema.

A pesar de que el convertidor es de 10 bits, se emplean solamente los 8 bits más significativos de la palabra binaria, a fin de facilitar su posterior procesamiento en la unidad central de procesos donde se ejecutan las aplicaciones maestras del sistema.

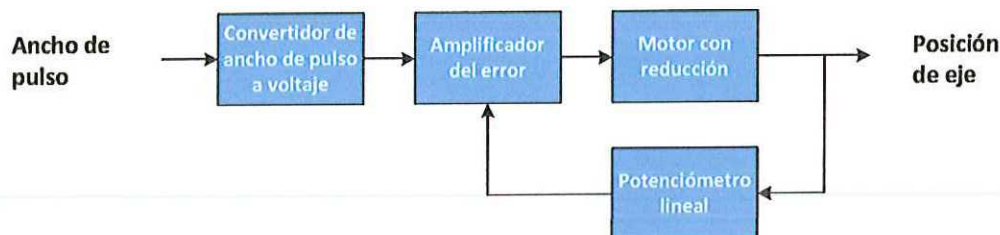
Para la configuración del módulo de conversión A/D en el microcontrolador, se procedió de la siguiente manera:

1. Configurar el módulo de conversión A/D:
2. Configurar los terminales necesarios como entradas analógicas y su voltaje de referencia.
3. Configurar el formato del resultado de la conversión justificado a la izquierda.
4. Seleccionar la procedencia del reloj del módulo de conversión A/D y el valor apropiado del tiempo de conversión por bit.
5. Seleccionar el canal analógico inicial.
6. Activar el módulo A/D.

### **5.2.3 Control de servomotores por modulación del ancho de pulso (PWM)**

El modelo del exoesqueleto esclavo es accionado por servomotores, los cuales permiten un posicionamiento preciso de su rotor dentro de su rango de operación. Esto se logra gracias al control electrónico interno del servomotor. El dimensionamiento y la selección de los servomotores empleados se detallan en el apartado 6.2.2 del próximo capítulo, sus especificaciones se pueden observar en el anexo F. En la figura 34 se presenta el diagrama de bloques general de un servomotor.

Figura 34. Diagrama de bloques de un servomotor dc.



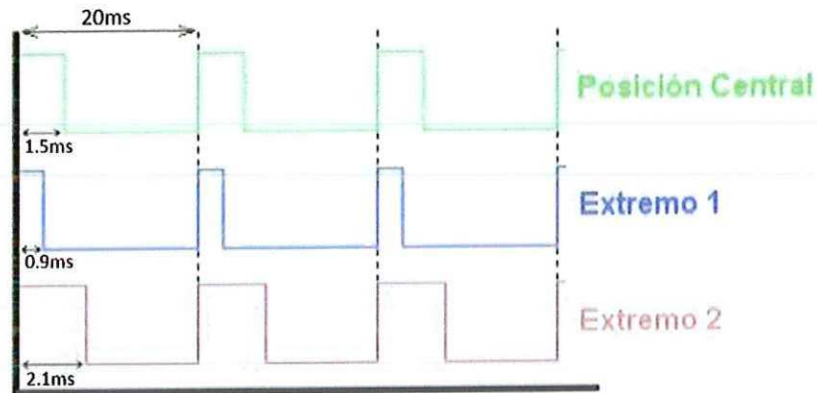
Fuente: El autor del proyecto

El funcionamiento del servomotor inicia con un pulso de control que es alimentado al convertidor de ancho de pulso a voltaje. Esto causa la carga de un capacitor a velocidad constante mientras el pulso permanezca en alto. Cuando el pulso cambia de estado, la carga del capacitor es alimentada a la salida de un amplificador seguidor de tensión. Esto produce un voltaje relacionado con la longitud del pulso aplicado. Este circuito convertidor está sintonizado de tal manera que produzca a la salida un voltaje útil con período entre 0.9ms y 2.1ms (ver figura 35). Luego la señal obtenida es restada con la señal de retroalimentación del potenciómetro acoplado al eje del motor, este error es alimentado a un amplificador inversor que tratará de minimizar la diferencia entre la entrada inversora y la no inversora. La salida de este restador es la señal usada para accionar el motor, si es positiva el motor gira en una dirección y si es negativa en la otra. De esta manera se logra el posicionamiento preciso del servomotor.

La señal de control de un servomotor debe ser una señal con ancho de pulso modulado (PWM). Es importante que dicha señal tenga la frecuencia de operación recomendada por el fabricante del servomotor, ya que no todas son las mismas y el uso de otras frecuencias causaría vibraciones y errores en el posicionamiento del eje.

Para el control de los servomotores se empleó entonces una tarjeta de la cual se disponía previamente, basada en un microcontrolador PIC16C621 desarrollada por Scott Edwards Electronics, capaz de controlar 8 servos simultáneamente. Esta tarjeta se encarga de traducir la posición deseada para un servo en una señal digital con ancho de pulso modulado a 50Hz. Las ventajas de esta tarjeta son su tamaño, su bajo consumo de potencia eléctrica, y su fácil operación. En el anexo D se detallan las especificaciones de la tarjeta y su manual de operación.

**Figura 35.** Ciclos útiles de los servomotores Hitec empleados.



Fuente: Modificado de [http://cfievalladolid2.net/tecno/cyr\\_01/robotica/sistema/motores\\_servo.htm](http://cfievalladolid2.net/tecno/cyr_01/robotica/sistema/motores_servo.htm)

### 5.3 FUENTES DE ENERGÍA

El sistema cuenta con circuitos electrónicos de baja potencia, por lo tanto la fuente de alimentación ideal para darle portabilidad al sistema son baterías recargables de Níquel metal hidruro. Lo importante en las fuentes a emplear es que se garanticen los niveles de tensión y de corriente que permiten el correcto funcionamiento de los diferentes circuitos y de los actuadores.

Teniendo en cuenta el voltaje de alimentación y el consumo de corriente máximo especificado por cada fabricante para los dispositivos y motores empleados, se puede proyectar un consumo energético para la selección de las baterías a emplear y la cantidad necesaria de éstas.

Respecto a los circuitos lógicos digitales, el sistema exigirá aproximadamente 148.9 mA, de los cuales 3mA son consumidos por los sensores, 15mA por el MAX232, 3.5mA por el receptor RWS-434, 19.4mA por el transmisor TWS-434A, 8mA por el regulador y 100mA por los microcontroladores. De esta manera el uso de pilas recargables tamaño AAA, cada una con una capacidad de 1300mAh y una tensión nominal de 1.2V, puede proporcionar una autonomía de aproximadamente 9 horas de funcionamiento. Además de esto las pilas otorgan ventajas relacionadas con el peso y volumen final del equipo. Para regular a una tensión fija de 5V se requieren por lo menos 5 pilas en serie para lograr un voltaje un poco superior al requerido.

Respecto al consumo de potencia del modelo de prueba que contiene 6 servomotores con consumo de corriente de 700mA cada uno en la prueba de eje bloqueado, se requiere entonces alrededor de 8 pilas recargables AA, que tienen un voltaje nominal de 1.2V y entregan 2100mAh, para lograr regular así a una tensión de 9V (voltaje de alimentación de los motores) con la corriente requerida.

## 6. DISEÑO MECÁNICO

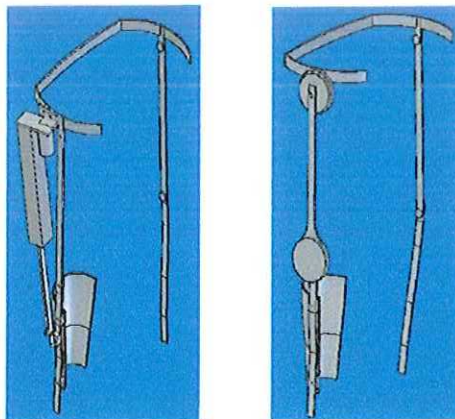
En este capítulo se describe el proceso de diseño mecánico del sistema. En una primera instancia se presentan una serie de modelos conceptuales que se desarrollaron para concebir la idea final del diseño del exoesqueleto, con las estructuras activa y pasiva interconectadas entre sí. Luego se presenta el diseño del modelo funcional construido para la validación del sistema completo, el cual consiste en una plataforma bípeda a pequeña escala. Se recuerda al lector, que este bípodo es experimental e irá desacoplado de la estructura pasiva ajustada a la pierna de la persona, pero se comportará como el sistema esclavo en reemplazo del exoesqueleto activo cuya construcción no se abarca en esta investigación.

### 6.1 DISEÑO CONCEPTUAL

En todo proceso de diseño se parte de una idea que poco a poco es desarrollada hasta realizar una selección de diseño. De esta manera, se menciona a continuación el recorrido que se realizó para el diseño de la propuesta final.

Para la selección del diseño y del mecanismo final, se contrastó entre dos propuestas desarrolladas (ver figura 36), cuya diferencia radica principalmente en el actuador y su respectivo funcionamiento.

**Figura 36.** Diseño conceptual de las propuestas iniciales.



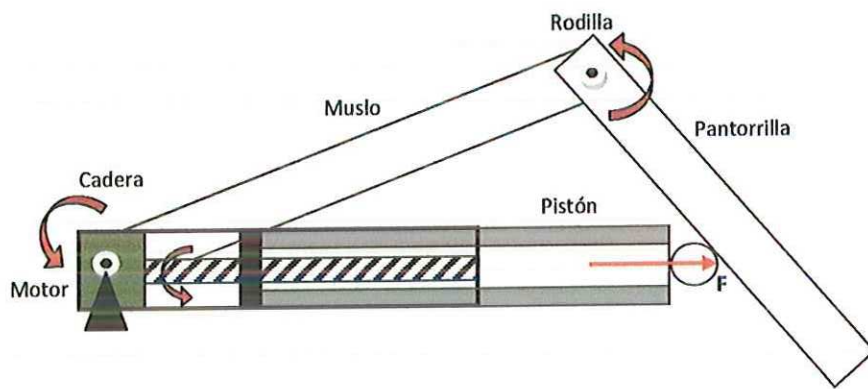
Fuente: El autor del proyecto. (a) con actuador lineal, (b) Con servomotores dc.



### 6.1.1 Diseño basado en actuador lineal:

La primera propuesta consistió en un mecanismo basado en actuador lineal compuesto por un tornillo de potencia acoplado en el eje de un motor dc. Al girar el motor el tornillo se genera el movimiento de la tuerca a la cual estaba acoplado el pistón que proporcionaría el empuje de la carga. La rotación de la pantorrilla y del muslo se lograría dejando la articulación de la cadera y la base del motor con libertad para girar. En la figura 37 se ilustra la idea.

Figura 37. Funcionamiento de la propuesta con actuador lineal.



Fuente: El autor del proyecto. (a) con actuador lineal, (b) Con servomotores dc.

La ventaja de este mecanismo es que con un motor de baja potencia ( $<90W$ ), se logra generar una fuerza de empuje que genera un gran torque ( $>25 Nm$ ) en las articulaciones. La desventaja se presenta en la limitación del movimiento y en la complicación del diseño, ya que como se observa en la figura 37, ambos grados de libertad están relacionados de manera dependiente, lo que conlleva a obtener una trayectoria de movimiento constante, siendo así el sistema inflexible para el caso de querer realizar terapias con movimientos diferentes en las piernas; para ello se necesitaría el diseño de un mecanismo extra para lograr el movimiento relativo en el punto donde se aplica la fuerza del pistón mecánico.

La opción de aprovechar el aumento de par mecánico que ofrece este mecanismo, y emplearlo en ambas articulaciones con un actuador independiente para cada una a fin lograr dar flexibilidad de movimiento, no fue considerado debido a que requeriría de una estructura bastante robusta y no sería ergonómico para la aplicación de rehabilitación.

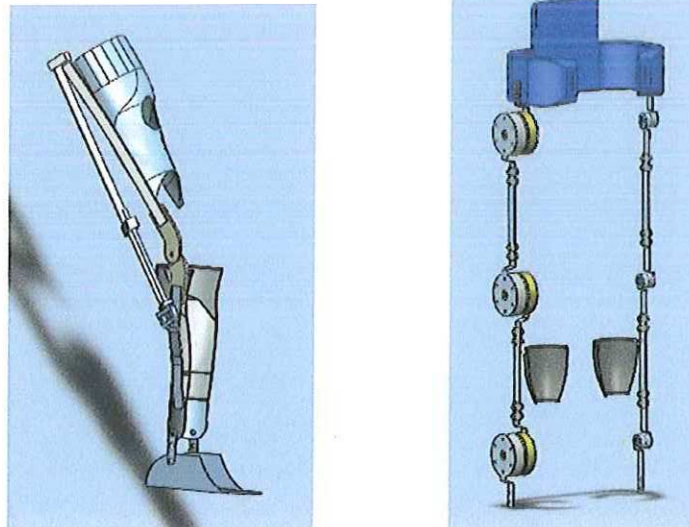
### 6.1.2 Diseño basado en servomotores

Dentro de la clasificación de los motores de corriente continua se encuentran los denominados *servomotores*, que están diseñados principalmente para el control preciso de la posición de su eje y para aplicaciones que requieren un alto par motor. El mecanismo propuesto en base a servomotores, tiene la ventaja principal de brindar independencia a cada articulación, además de facilitar las estrategias de control debido a que presentan un circuito interno para la regulación de posición del eje de acuerdo a la señal de consigna. Otro aspecto a resaltar es que su par mecánico alto se logra con una potencia eléctrica baja (<90W), ya que cuentan con una etapa reductora basada en trenes de engranes. El tamaño de estos actuadores es menor en comparación a la estructura necesaria para lograr estas prestaciones con motores convencionales o actuadores lineales. El funcionamiento de este mecanismo fue descrito con más detalle en el punto 5.2.3.

### 6.1.3 Selección del actuador

Teniendo en cuenta las alternativas de actuadores anteriores, se desarrollaron sus respectivos diseños detallados en 3D a fin de seleccionar la opción más apropiada de acuerdo a un conjunto de criterios ponderados. En la figura 38 se observan el diseño final propuesto para cada sistema.

Figura 38. Diseño final de cada propuesta.



Fuente: El autor del proyecto. (a) Con actuador lineal, (b) Con servomotores DC.

Empleando la metodología de comparación de soluciones por tablas de Pareto, se establecieron cinco criterios de evaluación de acuerdo a los intereses particulares del autor en cuanto al proceso de diseño, los cuales fueron contrastados entre sí con un valor de 0 a 10; de esta manera se determina el peso porcentual de cada criterio establecido. Para realizar la selección final, se evaluó cada propuesta en cada uno de los criterios con un valor de 0 a 10, luego se determinó el total de cada solución multiplicando cada valor asignado con el peso porcentual del criterio respectivo y sumando los resultados entre sí. En la tabla 3 se tabula el análisis de Pareto para la selección final del diseño.

**Tabla 3.** Criterios de selección de la solución. (A) Servomotores, (B) Actuador lineal.

CRITERIO	PESO [%]
Costos (C1)	12.72
Controlabilidad (C2)	17.27
Ventaja Mecánica (V)	19.09
Disponibilidad de Materiales (D)	19.09
Ergonomía (E)	22.72

**CONTRASTE DE CRITERIOS**

C1xC2	5-5
C1xV	3-7
C1xD	3-7
C1xE	3-7
C2xV	5-5
C2xD	6-4
C2xE	3-7
VxD	4-6
VxE	5-5
DxE	4-6

Solución	C1	C2	V	D	E	Total
A	3	9	9	7	7	6.6
B	3	7	9	5	7	5.8

← Solución seleccionada.

Fuente: El autor del proyecto

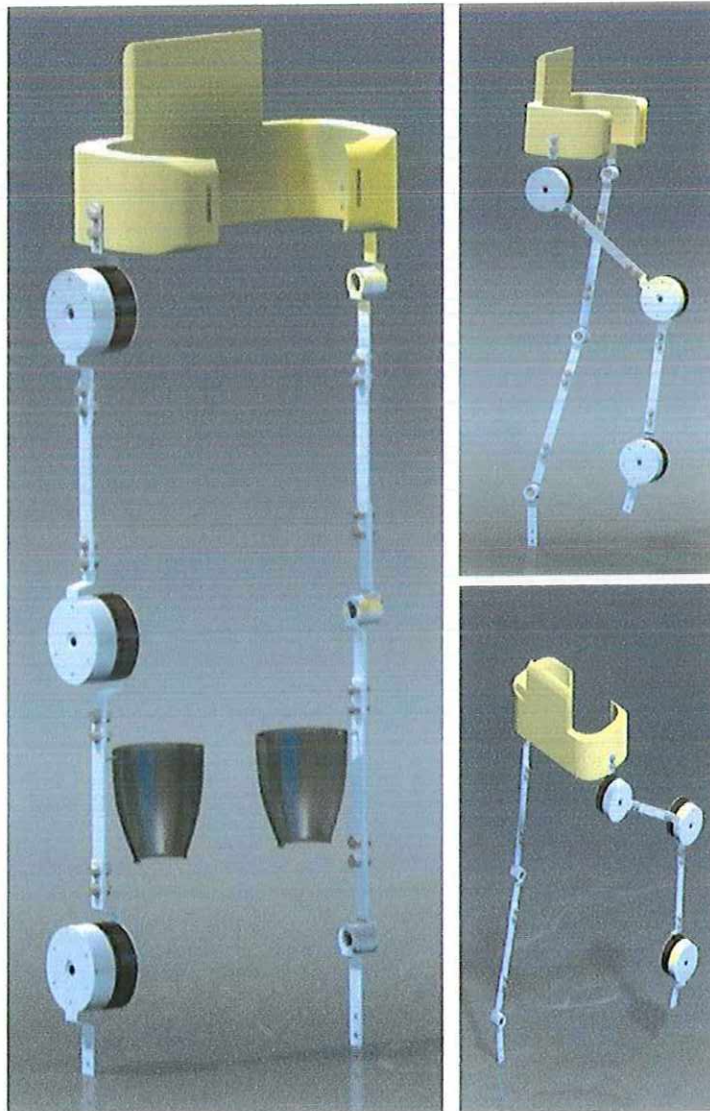
De esta manera se sustenta el uso de servomotores para la aplicación de rehabilitación de pierna que concierne a este proyecto. A continuación se detalla las partes que componen el mecanismo.

### 6.1.4 Diseño final del exoesqueleto

Para el diseño final del exoesqueleto se ha seleccionado el aluminio como material principal de la estructura pues a pesar de ser más costoso que otros metales como el latón y el acero, el aluminio es de menor densidad, de mayor facilidad para ser maquinado, es resistente a la corrosión, y comercialmente disponible.

Como se ha mencionado en capítulos previos, el exoesqueleto consta de: una estructura pasiva con sensores para grabar las trayectorias de una pierna, y de una estructura activa encargada de inducir dichos movimientos sobre la otra pierna con el uso de actuadores. En la figura 39 se observan algunas vistas del diseño final concebido, donde la pierna derecha es el sistema activo y la pierna izquierda el sistema pasivo. En el anexo E se presentan los planos acotados de cada pieza.

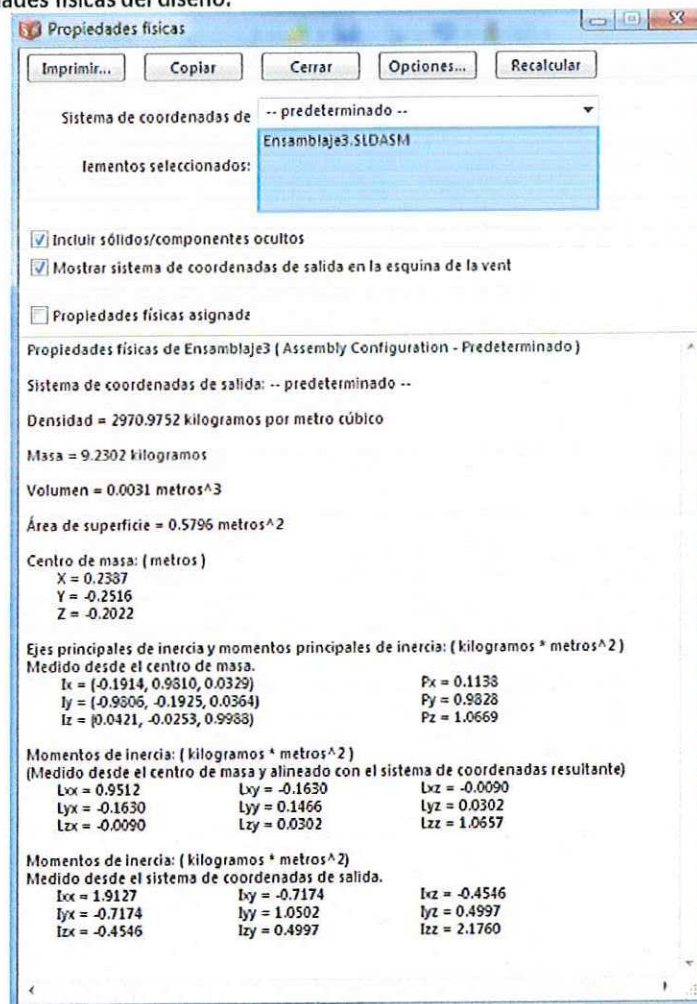
**Figura 39.** Imagen 3D del diseño desarrollado.



Fuente: El autor del proyecto

Para el análisis del diseño se empleó la herramienta SolidWorks con el fin de determinar las propiedades físicas del modelo que se especifican en la figura 40. Se puede observar que el sistema consta de dos cadenas cinemáticas de eslabones unidas entre sí por una estructura rígida no metálica en forma de cinturón que brinda estabilidad y soporte al usuario de la estructura. Cuantitativamente se verifica cómo el centro de masas del exoesqueleto está ligeramente desplazado hacia la pierna derecha del mecanismo por ser la que incluye los actuadores que proporcionan la mayor parte del peso de la estructura. Habiendo definido el material de cada elemento empleado, se calculó una masa de 9.23kg para el sistema, que sumado al peso de la persona para la cual se realizaron los cálculos, da un total de 70 kg aproximadamente.

Figura 40. Propiedades físicas del diseño.



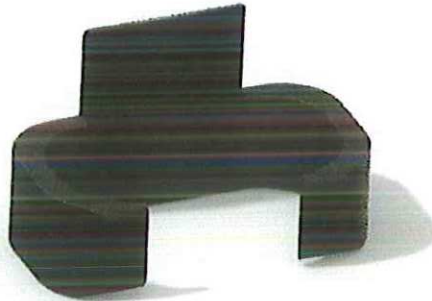
Fuente: El autor del proyecto. (Tomado de la ventana de análisis de la herramienta CAD empleada).

Es importante señalar que el exoesqueleto soporta la mayor parte del peso de la persona, pues al estar sujeto a ésta con el armazón que rodea la cintura, las fuerzas debidas a la gravedad se transmiten a la estructura metálica que se apoya en el suelo en su extremo inferior.

A continuación se desglosan las partes que componen cada cadena cinemática. Sus dimensiones están acordes a las del autor del proyecto, de 1.75 m de altura y 60 kg de masa. Las especificaciones y planos respectivos se presentan en el anexo E.

- **Cadera:** es el elemento que interconecta la estructura pasiva y la estructura activa del exoesqueleto. Se recomienda un material resistente y ligero como la fibra de carbono.

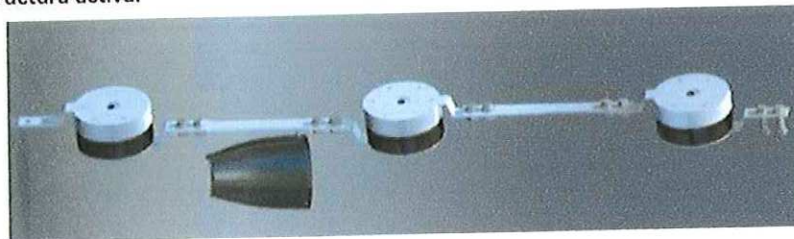
Figura 41. Cadera.



Fuente: El autor del proyecto

- **Estructura activa:** permite transmitir potencia del exoesqueleto a una pierna a través de los servomotores que accionan sobre el eje de rotación de cada articulación. Está compuesta por los motores, platinas de aluminio para la interconexión, ajustadores mecánicos (tornillos, tuercas y prisioneros), y una férula para el soporte de la pantorrilla.

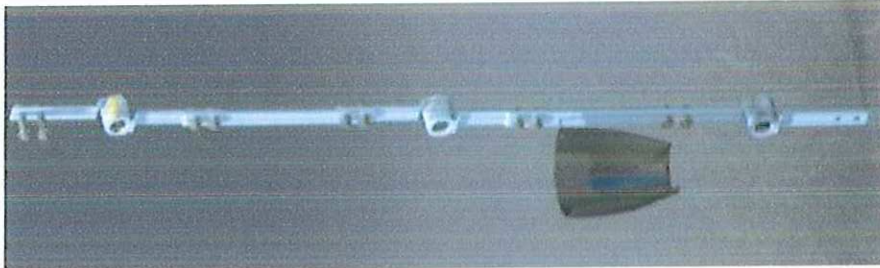
Figura 42. Estructura activa.



Fuente: El autor del proyecto

- **Estructura pasiva:** es la encargada de adquirir las señales analógicas que registran el movimiento de rotación de los segmentos de la pierna. Su estructura es muy similar que la parte activa, sólo difiere en que las juntas rotacionales no están formadas por motores sino por una rótula de aluminio con un potenciómetro lineal en su interior cuya base se ajusta a la platina de un segmento de la pierna y su eje a la platina del otro segmento de la pierna, así se logra el movimiento relativo.

Figura 43. Estructura pasiva.



Fuente: El autor del proyecto

Dentro del alcance planteado para este proyecto, se involucra sólo la construcción de la estructura pasiva. La validación de la respuesta se hará sobre una plataforma bípeda de menores dimensiones, con el fin de analizar los resultados del sistema maestro-esclavo, antes de realizar pruebas con una persona real. Aún así, se proponen en el anexo E, dos referencias de servomotores especiales para aplicar carga directamente sobre su eje sin mecanismo de transmisión o aumento de potencia. La selección de esas referencias se basa en los momentos máximos de cada articulación durante la marcha humana, referenciados por Winter en su libro *"Biomechanics and motor control of human motion"*.

Las especificaciones y planos respectivos a los elementos de cada ensamble descrito anteriormente se presentan en el anexo E.

## 6.2 MODELO DE VALIDACIÓN

A continuación se presenta la plataforma bípeda desarrollada para validar el sistema propuesto, así como el modelamiento mecánico para dimensionar los actuadores necesarios.

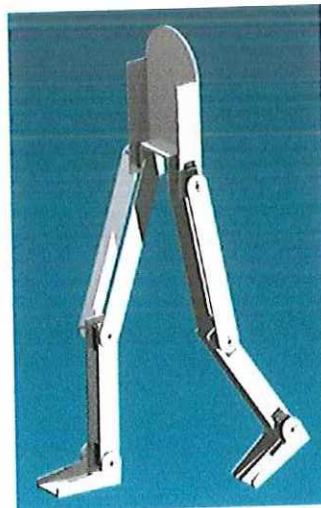
### 6.2.1 Modelado 3D

Para el diseño asistido por computador del bípedo de pruebas, se empleó el software especializado en diseño mecánico llamado Autodesk Inventor 2008 Professional. Se empleó la versión educativa, por ser gratuita y estar disponible en la comunidad de estudiantes en diseño e ingeniería en la página <http://student5.autodesk.com/>. Las dimensiones de los segmentos de la plataforma fueron escaladas en un factor de 1 : 2.3 según las proporciones de los segmentos reales de la pierna calculadas en el apartado 4.1.

Las dimensiones y materiales empleados para la construcción del modelo de validación se fundamentaron en las características de potencia mecánica de los servomotores a los cuales se tuvo acceso en el mercado local. Estos motores son de la marca Hitec. Se emplearon dos referencias diferentes cuya justificación se plantea en el apartado 6.2.2: para la articulación de la cadera se empleó el servomotor HS-475HB, mientras que para la rodilla y el tobillo se emplearon los HS 311. Las dimensiones de estos servomotores son exactamente iguales, la diferencia radica en sus características constructivas internas.

En la figura 44 se presenta la imagen espacial del modelo funcional construido, el cual emula los segmentos de las piernas humanas: pies, pantorrillas y muslos. También presenta una especie de torso en el cual se alojan los circuitos electrónicos. Cabe señalar que se pretende mover el mecanismo según una señal maestra más no que la plataforma presente equilibrio dinámico, por ello es necesario sujetarla por el torso.

Figura 44. Diseño 3D de la plataforma bípeda experimental.



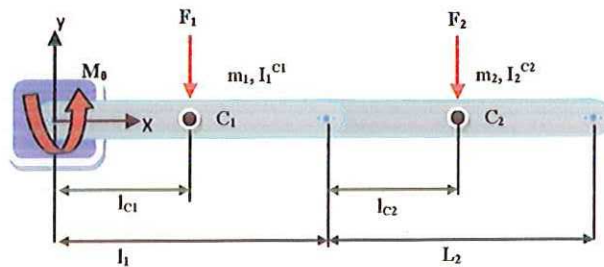
Fuente: El autor del proyecto



## 6.2.2 Dimensionamiento y selección de los servomotores

Para seleccionar los motores a emplear se tuvo en cuenta un caso extremo de posición de la pierna de la plataforma, el cual nunca ocurre durante la marcha pero permite incluir entonces cierto factor de seguridad al calcular un torque un poco mayor al requerido. Teniendo en cuenta la figura 45 del modelado de una pierna con dos eslabones, y las masas y longitudes del bípedo, se calcula el momento de fuerzas alrededor de la cadera para una posición horizontal de la pierna donde el ángulo del muslo y de la pantorrilla con respecto a la horizontal es cero.

Figura 45. Diagrama de cuerpo libre para el dimensionamiento de motores



Fuente: El autor del proyecto

Para la cadera:

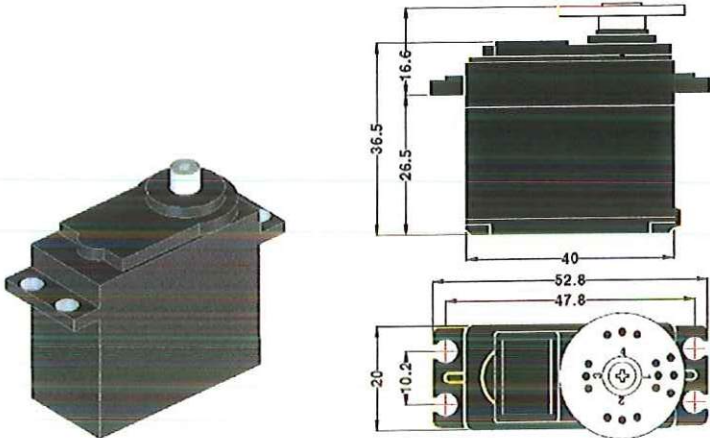
$$M_0 = l_{c1}F_1 + l_{c2}F_2 = (0.18\text{m} \times 0.047\text{kg} + 0.34\text{m} \times 0.051\text{kg}) \times 9.8 \frac{\text{m}}{\text{s}^2} = 0.2528 \text{ Nm}$$

Para la rodilla:

$$M_1 = l_{c2}F_2 = (0.18\text{m} \times 0.047\text{kg}) \times 9.8 \frac{\text{m}}{\text{s}^2} = 0.0829 \text{ Nm}$$

De esta manera se debe asegurar que el motor de la cadera y de la rodilla proporcionen un par aplicado mayor o igual a  $M_0$  y  $M_1$ , respectivamente. Por lo tanto, los motores seleccionados Hitec HS-475HB y HS-311, con torques de 0.4315 Nm y 0.2942 Nm operando a 4.8V, satisfacen los requerimientos calculados. En la figura 46 se observa el modelado 3d de estos servomotores y sus respectivas dimensiones. Las especificaciones técnicas se presentan en el anexo F.

Figura 46. Diseño 3D de los servomotores.



Fuente: El autor del proyecto

## **7. DESARROLLO DEL SOFTWARE**

El software desarrollado en este proyecto consta de 3 secciones. La primera de ella es la encargada de la adquisición y digitalización de las señales del exoesqueleto pasivo acoplado a la pierna sana, lo cual está implementado directamente en la unidad local con la programación del microcontrolador. La segunda sección es una interfaz de usuario que permite realizar el cálculo de los ángulos articulares a partir de la información binaria recibida, la presentación de los resultados en gráficas y animaciones, y el control de los servomotores de la plataforma bípeda para su validación desde un computador. La tercera sección consiste en modelos y simulaciones en 3D de la marcha humana y de los movimientos que se registren con el exoesqueleto pasivo. Estas aplicaciones se trabajaron principalmente en MATLAB R2007b.

En las siguientes secciones de este capítulo se explican los detalles de cada nivel de programación, los cuales fueron desarrollados de manera modular para asegurar el funcionamiento por etapas, luego se explica la interfaz que los integra en una sola aplicación.

### **7.1 INTERFAZ DEL SISTEMA LEGXOS**

Los ángulos relativos entre segmentos de una pierna son adquiridos inicialmente por el microcontrolador PIC16F873, quien digitaliza las señales analógicas a un formato binario de 8 bits y las transmite al computador para que las traduzca en un valor angular en formato decimal y realice las acciones necesarias para accionar los servomotores. El programa encargado de gestionar estas tareas contiene rutinas de inicialización, encargadas de configurar los diversos periféricos a usar en el microcontrolador, otras de comunicación para establecer el intercambio de datos en modo serial con el computador, quien ejecuta las funciones de cálculos, gráficos de las señales, simulaciones, accionamiento de motores y manejo de históricos y base de datos de pacientes.

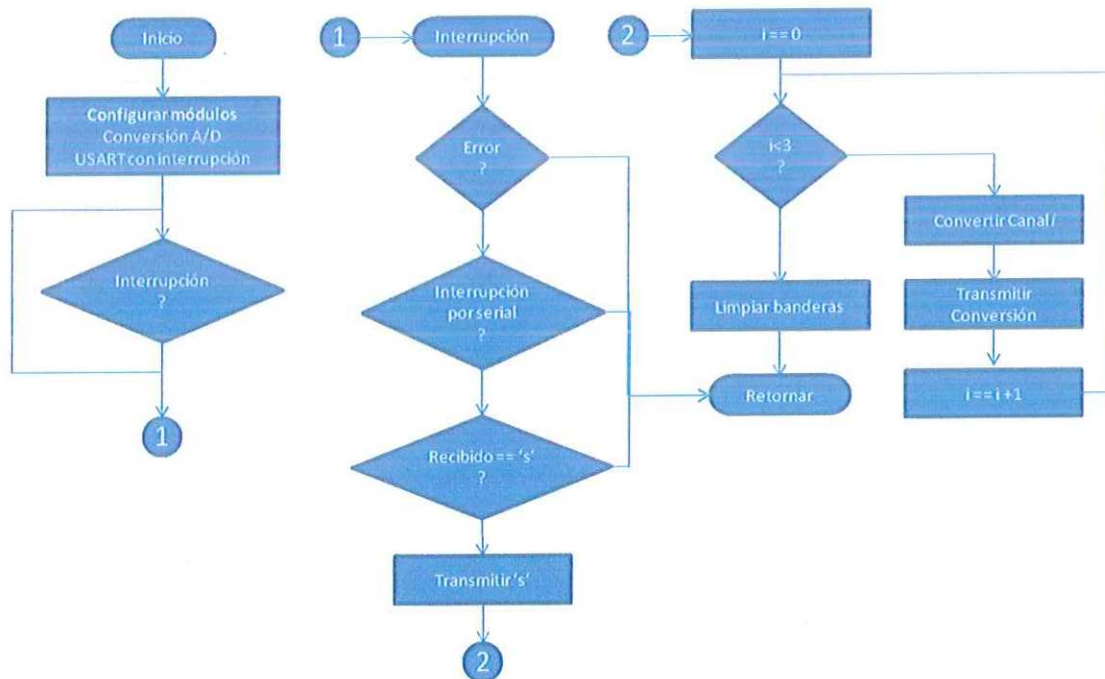
A continuación se presenta el diagrama de flujo de operación del microcontrolador y la interfaz desarrollada en Matlab.

### 7.1.1 Programación de la unidad local basada en PIC16F873A

Como se observa en la figura 47, la primera operación que se realiza es la rutina de configuración e inicialización de los registros necesarios para hacer uso del módulo de comunicación serial USART (Universal Synchronous Asynchronous Receiver Transmitter) y del módulo de conversión análogo-digital que presenta el microcontrolador seleccionado. Los parámetros a configurar fueron detallados en la sección 5 relativa al diseño electrónico del sistema.

Posterior a la inicialización, el microcontrolador entra en un ciclo infinito esperando que ocurra una interrupción. Se verifica que dicha interrupción haya sido por recepción serial y no por algún otro módulo o error. El siguiente paso es comparar el valor recibido con el código ASCII de la letra 's', la cual se seleccionó para solicitar un muestreo de las señales desde el pc. Si el dato recibido resulta ser dicho carácter, entonces se procede a iniciar la conversión y transmisión de los 3 canales analógicos que se están monitoreando, referentes al ángulo de la cadera, rodilla y tobillo. Al finalizar el envío del último paquete, se regresa al estado de espera en un ciclo infinito.

Figura 47. Diagrama de flujo del microcontrolador para adquisición de datos.

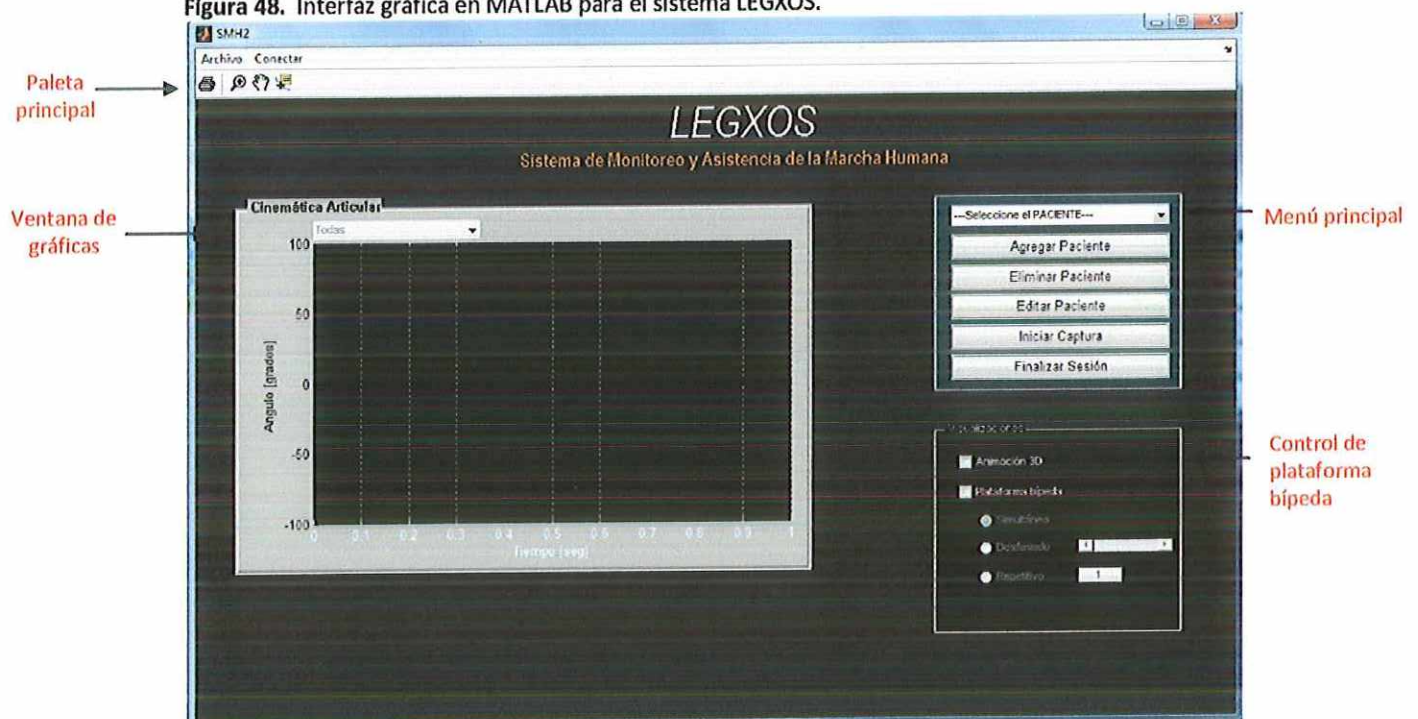


Fuente: El autor del proyecto

## 7.1.2 Interfaz Gráfica de Usuario

A pesar de que el microcontrolador es el centro de la adquisición de las señales, es necesario traducir dichos valores binarios en un formato decimal que permita realizar cálculos, visualizar la información y archivar los datos para cada sesión de monitoreo. Para ello se desarrolló una interfaz gráfica de usuario que cumple con los requerimientos establecidos en el capítulo 3. A continuación se detallan las operaciones que se ejecutan con la interfaz.

Figura 48. Interfaz gráfica en MATLAB para el sistema LEGXOS.



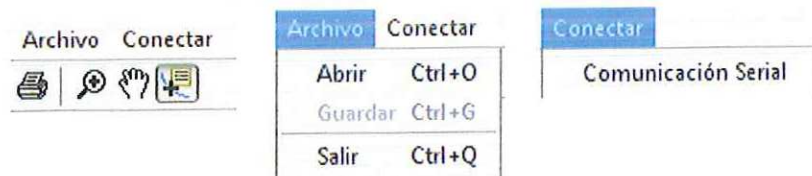
Fuente: El autor del proyecto

En la figura 48 se puede observar la pantalla inicial de la interfaz. En la parte superior derecha de la imagen aparece el menú principal que permite gestionar la base de datos de pacientes y dar inicio o fin a la sesión de monitoreo y reproducción. En la parte derecha inferior se observa un cuadro de selección con las opciones de visualización de la respuesta del sistema, sea por animación 3d o por accionamiento de la plataforma bípeda. En la parte central se encuentra el gestor de gráficas. En la esquina superior izquierda se encuentra la paleta principal de funciones, que permite abrir históricos, guardar resultados, manipular los gráficos en pantalla, verificar la conexión con la unidad local de procesamiento y el estado del puerto serial.

Paleta principal:

Esta paleta consiste en una serie de botones y submenús que permiten interactuar con la interfaz (ver figura 49). Está conformada por los menús *Archivo* que permite abrir y guardar archivos de extensión *.mat*, que contengan los campos de información relativos a esta interfaz, y *Conectar* que permite verificar el estado de conexión del puerto serial. También contiene los botones *Imprimir*, *Zoom*, *Pan* y *Data Tip*, especiales para interactuar con los resultados gráficos, que se comprimen gráficamente mientras la sesión aumenta en tiempo, por lo tanto permiten dar una visión más detallada de alguna zona o valor en interés.

Figura 49. Paleta principal de la interfaz.

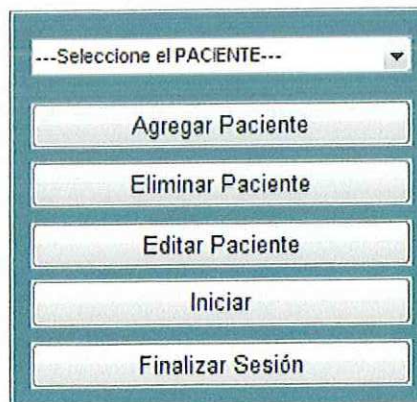


Fuente: El autor del proyecto

Menú principal:

Consiste en una serie de botones que permiten la gestión de pacientes que serán asistidos con este sistema (ver figura 50).

Figura 50. Menú principal de la interfaz.



Fuente: El autor del proyecto

El menú principal presenta varias opciones. El botón *agregar* ingresa a un nuevo paciente en el sistema, lo cual es un requisito si se desea realizar una sesión de análisis. También permite *eliminar* el paciente o *editar* su información. Al registrar a un nuevo paciente se solicitan los datos que se ilustran con la figura 51; estos datos son los mínimos necesarios para tener la información de contacto del individuo y para realizar los cálculos necesarios de centros de masas, momentos de inercia, longitudes de segmentos, entre otros. Si los datos ingresados son correctos al formato del campo, aparecerá un aviso de aceptación del nuevo paciente, de lo contrario, no se podrá registrar en el sistema. Al ser satisfactorio el registro, se crea en el directorio raíz del software, una carpeta con el nombre de registro del paciente, con el fin de que se guarden en ella la información de las diferentes sesiones que realice dicho paciente.

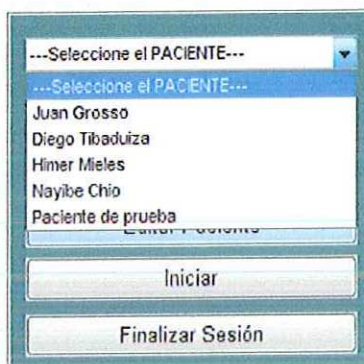
Figura 51. Menú desplegable para el registro de nuevos pacientes.

The image shows two windows from a software application. The larger window on the left is titled "Datos del Paciente" and contains several input fields: "Nombre" (a wide text box), "Edad" (a small text box), "Teléfono" (a small text box), "Masa [kg]" (a small text box), "Altura [m]" (a small text box), "C\_muslo [m]" (a small text box), and "C\_pant [m]" (a small text box). At the bottom of this window are two buttons: "Registrar" and "Cancelar". To the right of this window is a smaller dialog box titled "BD" with a message that says "El paciente ha sido registrado" and an "OK" button.

Fuente: El autor del proyecto

Otra opción que presenta este menú es el desplegable *Seleccionar paciente* el cual es necesario de ejecutar cada vez que se desee agregar, eliminar, editar o iniciar sesión. En él se muestra la lista de pacientes registrados en el sistema. En la figura 52 se observa un ejemplo de lista con pacientes previamente registrados.

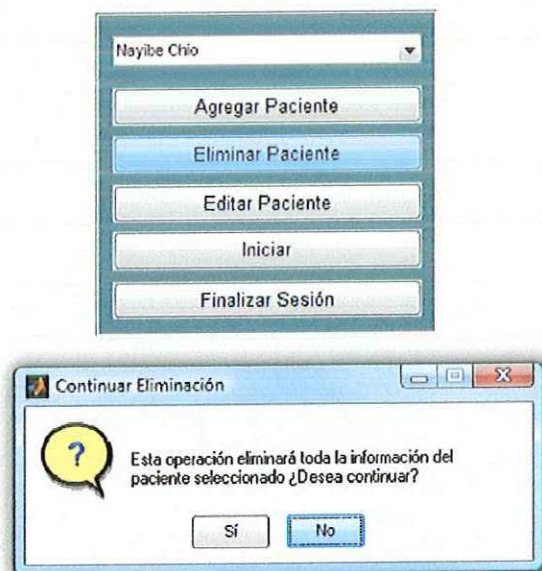
**Figura 52.** Lista de pacientes registrados.



Fuente: El autor del proyecto

Es importante señalar que la interfaz también realiza la administración de mensajes de alerta y error a fin de facilitar el uso correcto del sistema. En la figura 53 se presentan los avisos relativos a la acción eliminar de este menú principal.

**Figura 53.** Mensajes de alerta en la opción eliminar.

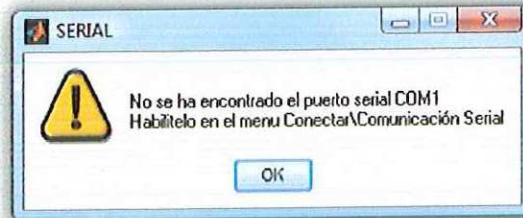


Fuente: El autor del proyecto



Por último, en este menú aparecen los botones *iniciar* y *finalizar* sesión. Al querer iniciar una sesión el sistema primero verifica el estado de la comunicación, enviando y esperando el eco de una señal que sirve de sincronía para iniciar la descarga de los datos monitoreados. Si no se recibe la respuesta esperada dentro del tiempo estipulado, entonces se despliega un mensaje de error para notificar el problema (ver figura 54).

Figura 54. Mensaje de alerta al no establecer comunicación.

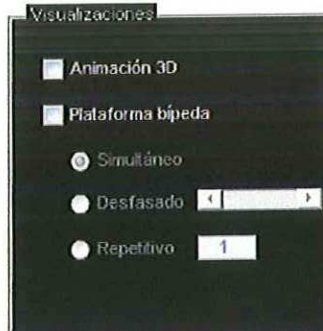


Fuente: El autor del proyecto

### Menú Visualizaciones:

Este menú permite seleccionar el tipo de salida para observar la respuesta del sistema (ver figura 55). Se presentan dos opciones principales: *Animación 3D* y *Plataforma bípeda*. La primera de ellas permite reproducir en un entorno 3D simultáneamente el movimiento que se está monitoreando de la pierna. Por otro lado la segunda opción permite elegir el modo de operación que se desea ejecutar sobre la plataforma bípeda real, éste puede ser en *simultáneo* (la estructura física se accionará al tiempo que se captura el movimiento), *desfasado* (el accionamiento de la estructura estará desfasado  $t$  segundos del momento de captura del movimiento) y *repetitivo* (permite reproducir  $n$  veces deseadas un patrón de movimiento previamente registrado).

Figura 55. Menú Visualizaciones y modos de operación.

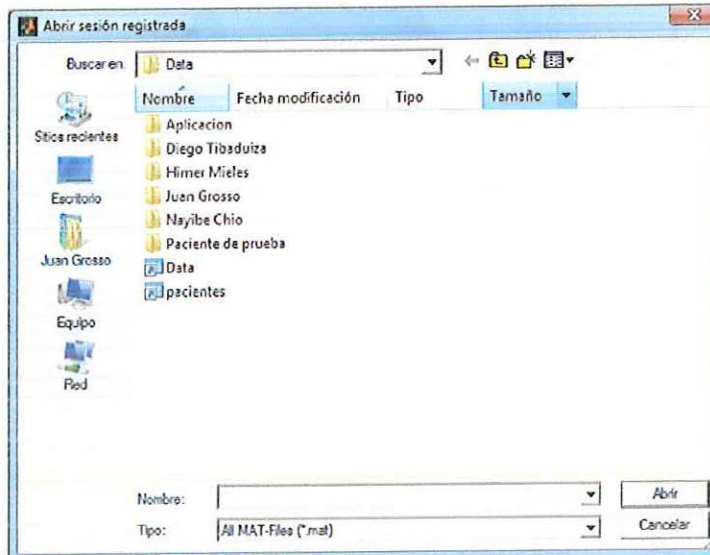


Fuente: El autor del proyecto

### Submenú Abrir:

Permite abrir archivos guardados previamente con esta interfaz. Al ejecutar esta opción se verifica que el archivo con extensión *.mat* seleccionado, contenga los campos asociados al archivo cuya estructura patrón, denominada *Data*, fue creada desde la instalación del software.

Figura 56. Opción Abrir archivo.



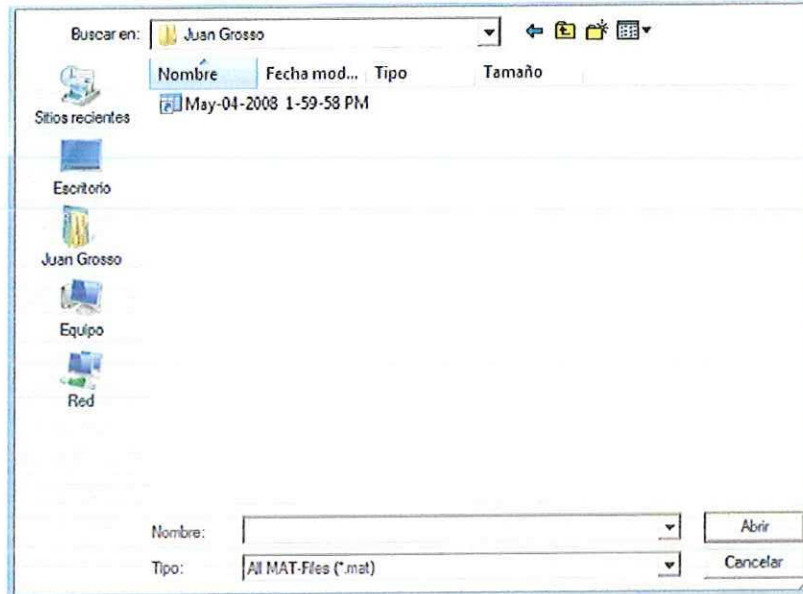
Fuente: El autor del proyecto

En la figura anterior también se puede notar la existencia de una carpeta particular creada para cada paciente registrado, que contiene el histórico de las diferentes sesiones de terapia que se haya realizado. El archivo *pacientes.mat* que se observa en la figura previa, es el encargado de contener la información de contacto de cada paciente, podría decirse que es una base de datos, desarrollada no en un servidor de bases sino programáticamente con el manejo de estructuras, campos y archivos.

### Submenú Guardar:

Este submenú inicia deshabilitado y solo se activa si se ha detenido una sesión de adquisición de datos. Al aceptar guardar la información, el directorio raíz se convierte en la carpeta del paciente analizado y se propone un nombre automático de archivo, formado por la fecha y la hora de la prueba realizada con el fin de poder organizar las sesiones que se realicen. De todas formas es posible aplicar el nombre que se desee.

Figura 57. Opción Guardar archivo.

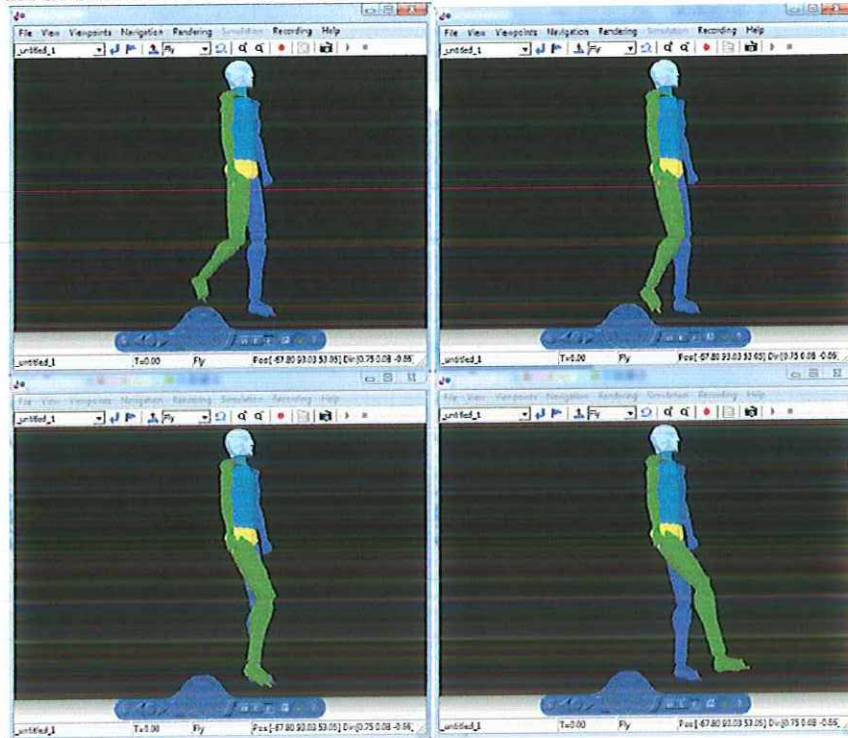


Fuente: El autor del proyecto

### 7.1.3 Simulaciones en 2D y 3D

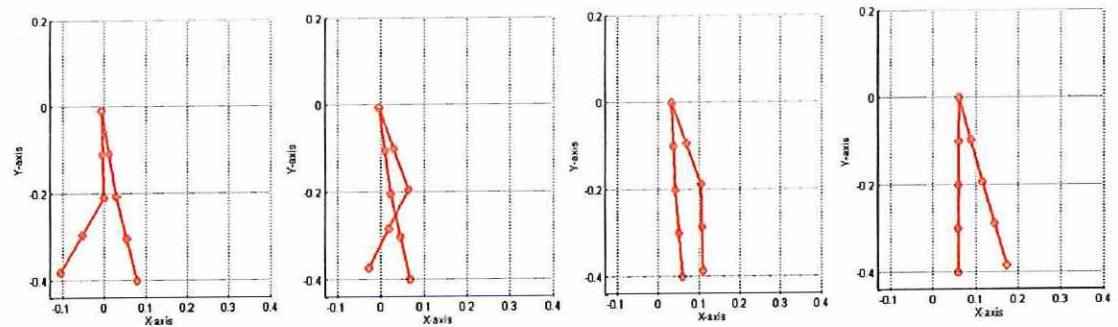
Con el fin de observar más fácilmente los movimientos registrados y poder tener una mejor idea de lo que las gráficas presentan, se programaron unos primeros modelos gráficos en 2D y 3D, que pueden ser animados con retardo o en tiempo real con los datos que se están obteniendo. Para ello se empleó la herramienta de Realidad Virtual de MATLAB, que permite crear entornos virtuales sencillos, o importarlos después de generarlos en software especializados en diseño gráfico en 3d como por ejemplo 3D Studio Max de la casa Autodesk. También se empleó la herramienta mencionada en el capítulo 4, SimMechanics de Simulink, que permite realizar la programación de los modelos físicos matemáticos y de los controles necesarios. A continuación se muestran algunos resultados con los entornos mencionados en la fase de balanceo de la pierna.

Figura 58. Fase de balanceo en entorno de realidad virtual.



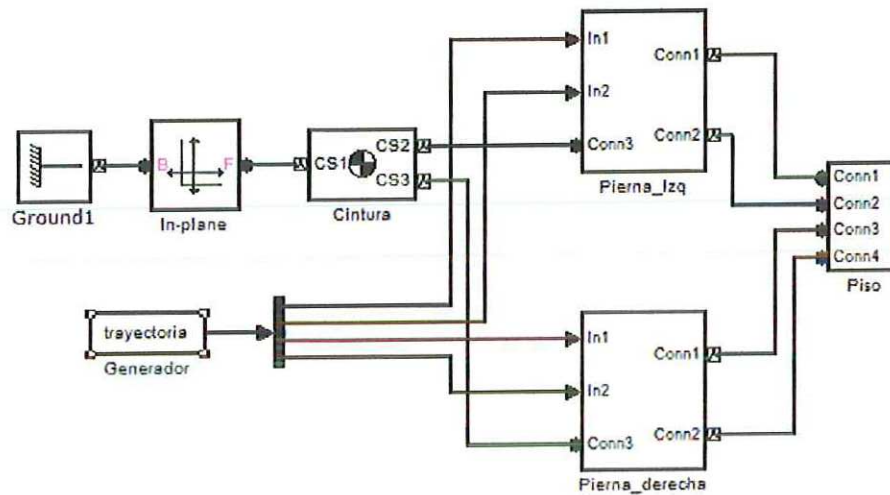
Fuente: El autor del proyecto

Figura 59. Fase de balanceo en entorno de SimMechanics.



Fuente: El autor del proyecto

Figura 60. Modelo de cuerpos rígidos en SimMechanics para la marcha.



Fuente: El autor del proyecto

Donde,

- *Ground 1*: es el sistema inercial de referencia situado en el punto  $[x,y,z]=[0,0,0]$ .
- *In-plane*: representa los grados de libertad del cuerpo rígido denominado cintura. Es una articulación prismática de dos ejes  $(x,y)$ .
- *Cintura*: es el bloque de configuración del cuerpo rígido que representa la cintura.
- *Pierna-izq*: este bloque representa el subsistema de la pierna izquierda. Dentro de él se configuran las articulaciones de rotación de la cadera y de la rodilla, así como los elementos sensores y actuadores de tales juntas. Contiene los bloques de cuerpo rígidos del sistema muslo-pantorrilla para la pierna izquierda.
- *Pierna-der*: este bloque representa el subsistema de la pierna derecha. Dentro de él se configuran las articulaciones de rotación de la cadera y de la rodilla, así como los elementos sensores y actuadores de tales juntas. Contiene los bloques de cuerpo rígidos del sistema muslo-pantorrilla para la pierna izquierda.
- *Piso*: este bloque permite simular el piso y las fuerzas de reacción. Está fundamentado en los trabajos realizados por O.Vele [12] en el desarrollo de un robot humanoide en Matlab.
- *Trayectoria*: este bloque representa las coordenadas de articulares que describen el movimiento de cada pierna, contiene las coordenadas generalizadas para cada pierna, es decir, la variación angular tanto para la cadera y rodilla izquierda como derecha.

En la figura 60 se presenta el modelo generado en Simulink para la simulación de la marcha humana. Los códigos fuentes de cada subsistema se pueden revisar en el anexo G.

## 8. PRUEBAS Y RESULTADOS

Con el fin de evaluar los diferentes desarrollos que se plantearon en este proyecto de grado, se realizaron algunas pruebas de desempeño del sistema, tanto pruebas de hardware como de software.

### *Pruebas con la comunicación serial*

La primera prueba que se realizó fue comprobar que se establecía una comunicación correcta entre el computador central y los microcontroladores locales. Para ello se diseñó un programa simple en el cual, la unidad local debía retornar el eco de un carácter enviado desde el Hyperterminal del computador. El resultado en este aspecto fue el esperado, por lo tanto se procedió a realizar la misma prueba con Matlab, con el fin de verificar el comportamiento de la lectura y escritura del puerto serial en dicho software.

Se programó un ciclo repetitivo en el cual se escribe y se lee el puerto continuamente con el fin de evaluar la estabilidad de la comunicación. Se observó que la función interna de Matlab para la escritura del puerto serial presenta problemas en momentos inesperados, lo que ocasiona que se detenga el programa en ejecución. Se intentó solucionarlo variando la rata de baudios y otras propiedades del puerto serial pero ocurría lo mismo. Finalmente, se contactó a un técnico de Matlab para indagar por el extraño comportamiento de la función de escritura, ésta persona confirmó que era un error interno del software que estaría resuelto en las próximas versiones de Matlab. De tal manera que el sistema puede presentar paradas inesperadas en algún momento, y se debe volver a dar la orden de escritura.

Cabe anotar que estas pruebas se realizaron con comunicación cableada, de tal manera que para dar mayor flexibilidad se implementaron dos opciones inalámbricas. La primera se realizó empleando módulos bluetooth, *BlueRS232*, la comunicación fue exitosa trabajando con un computador de escritorio como sistema central, pero se encontró un problema al trabajar con computadores portátiles. Esto se debe a que se requiere de un cable convertidor de RS232 a USB, puesto que los portátiles no presentan el conector DB9 del protocolo serial, lo que ocasiona que algunas tramas de datos necesarias para el control de flujo de los módulos *BlueRS232* entren en conflicto y se inhabilite el puerto serial en uso. De esta manera se procedió a implementar comunicación inalámbrica con módulos de radio frecuencia, cuya referencia es *TWS-434* y *RWS-434*, con ellos se resolvió el inconveniente de los portátiles ya que sólo se requiere una línea para el envío de los datos digitales. En este caso la problemática se debe es a interferencias y a pérdida de bits durante el envío. Inicialmente se emplearon antenas convencionales que cumplieran con

la distancia recomendada por el fabricante de los módulos RF, pero fue necesario adquirir antenas especiales para esas aplicaciones, diseñadas para la misma frecuencia de dichos módulos. Fue necesario a su vez, implementar una etapa comparadora a la salida lineal del receptor con el fin de establecer el umbral de niveles lógicos antes de enviar los datos al microcontrolador, de esta manera se aseguran los valores de los bits recibidos ya que pierden su amplitud en la transmisión y se afectan por la distancia. El rango de alcance obtenido fue de unos 6 m aproximadamente.

### ***Pruebas con la conversión análoga-digital***

En una primera instancia el circuito de adquisición de datos se implementó con un microcontrolador PIC16F877A pero se observaron comportamientos erróneos en la respuesta del convertidor análogo digital ya que existía solapamiento de la señal convertida de un canal sobre la del canal siguiente, lo cual ocasionaba sobre niveles en las señales obtenidas y por tanto se indicaban desplazamientos no realizados.

Este efecto se le conoce con el nombre *ghosting*. Se analizaron sus posibles causas, y se tomaron las medidas pertinentes para evaluar cambios, entre las que se mencionan: se verificó que la impedancia del sensor fuese baja ( $<10 \text{ kohm}$ ), se redujeron los tiempos de muestreo y se ampliaron los retardos en el cambio de canal con el fin de asegurar el tiempo de asentamiento del convertidor, se implementó una etapa de seguidor a la salida del sensor para el acople de impedancias a la entrada del microcontrolador, se usaron filtros rc, entre otras. Ninguna de las pruebas dio resultados positivos por lo que se decidió intentar realizar la conversión análoga digital en un convertidor exclusivo para ese fin. El resultado mejoró pero se obtenían picos en momentos inesperados, así que finalmente se decidió cambiar la referencia del microcontrolador a un PIC16F873A en el cual se implementó el mismo programa. Al realizar las pruebas observamos que se obtuvo una señal continua sin alteraciones ni solapamientos entre canales, tal como se esperaba según lo establecido en el capítulo de diseño electrónico. En la figura 61 se comparan la señal errónea con la señal correcta.

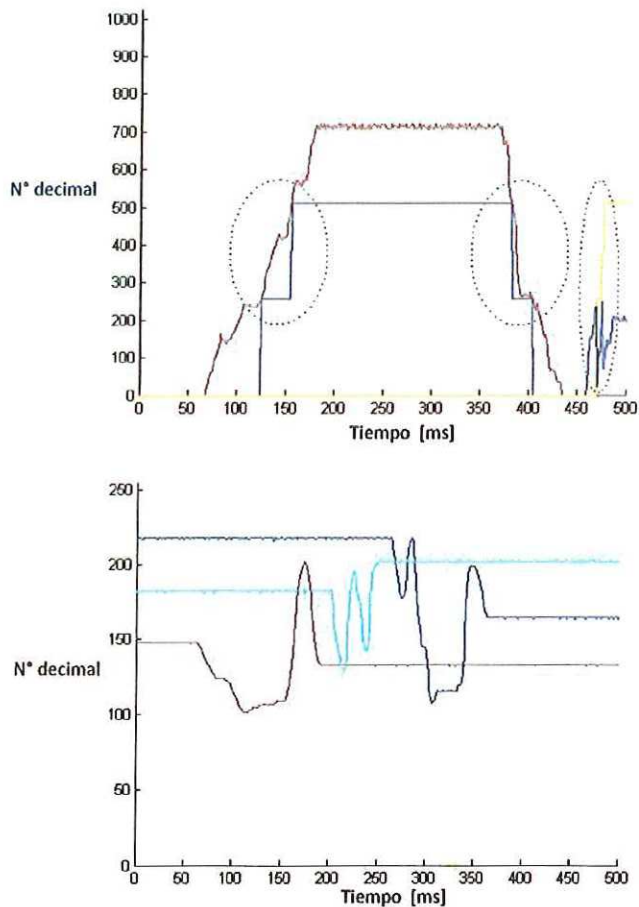
### ***Prueba para calibración de datos***

Luego de asegurar una recepción estable de datos, se verificó la concordancia de éstos y se comprobó que el rango de datos posibles de monitorear fuera correctamente interpretado en el sistema. En una primera prueba se graficó el valor del número binario recibido por el puerto serial pero en formato decimal, la respuesta esperada son valores dentro del intervalo de 0-255 desde un ángulo de  $0^\circ$  a uno de  $310^\circ$  que es el barrido

máximo de los potenciómetros empleados como sensores. Gracias a la linealidad de los sensores empleados cuya ecuación característica fue determinada en el capítulo 4, la calibración del dato se logró sólo con multiplicar el valor adquirido por un factor de 310/255, que permite pasar el numero decimal del dato binario recibido a su relativo en grados para determinar el ángulo.

En este punto se validó el sistema de forma computacional en el entorno de realidad virtual con el fin de animar el humanoide directamente con las señales obtenidas del exoesqueleto. Se corroboró que el programa fuese capaz de seguir la trayectoria monitoreada en tres condiciones: en simultáneo, con retardo y repitiendo el movimiento  $n$  veces deseadas.

**Figura 61.** Comparación de respuestas del convertidor A/D.



Fuente: El autor del proyecto. (*Superior*) Efecto de solapamiento en la señal. (*Inferior*) Señal corregida.

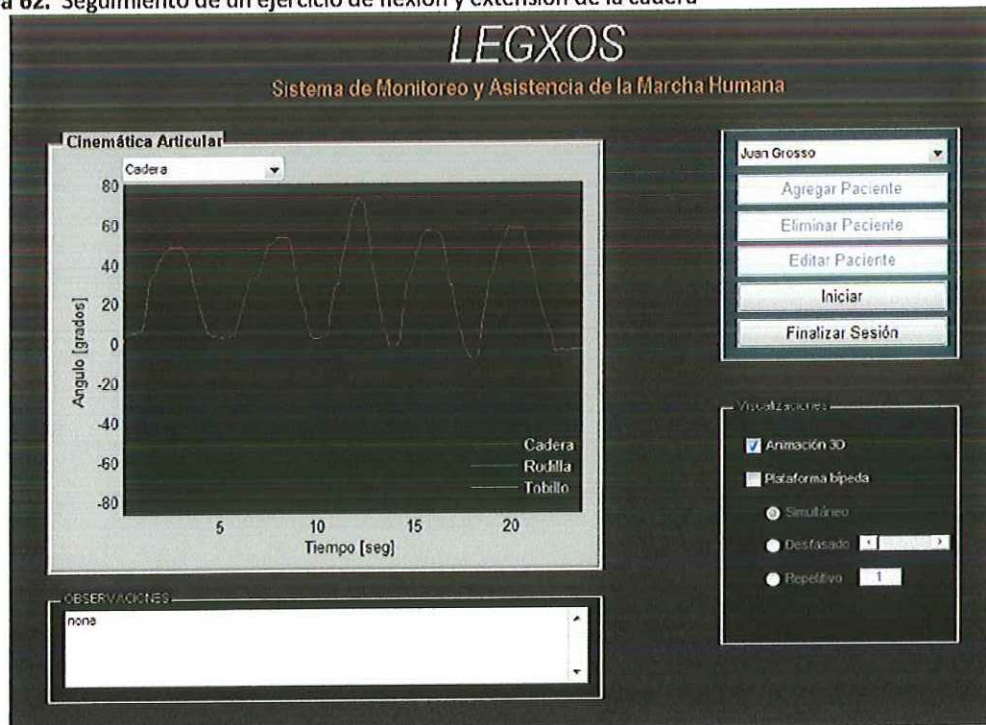


### **Prueba de accionamiento de la plataforma bípeda**

La última prueba realizada fue verificar que se accionaran los servomotores para realizar los movimientos tal como se validó con el modelo virtual. En este aspecto se obtuvo una respuesta adecuada al trabajar articulación por articulación, pero al energizar todo el sistema se observó que en algunos momentos la plataforma bípeda de pruebas presentaba vibración e inestabilidad, debido a que la señal que se adquiere no está siendo filtrada digitalmente y presenta unas leves oscilaciones alrededor de la posición real.

Por otro lado se intentó generar un patrón de marcha a partir de una generación matemática de trayectorias aplicando cinemática inversa en el espacio cartesiano para luego interpolar las coordenadas articulares obtenidas, pero es necesario realizar ajustes al diseño del bípodo con el fin de incluir control de equilibrio ya que en el caso de este proyecto se asume que la persona es quien brinda el balance al sistema exoesqueleto-paciente.

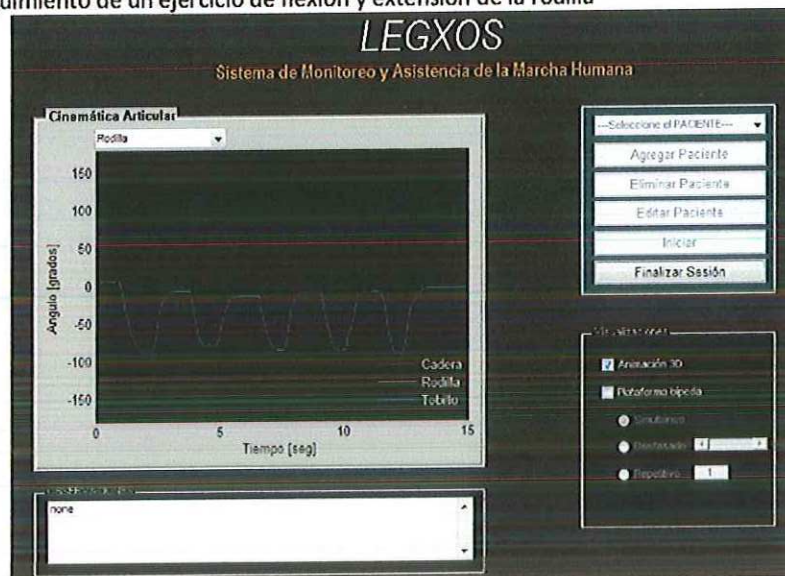
**Figura 62.** Seguimiento de un ejercicio de flexión y extensión de la cadera



Fuente: El autor del proyecto.

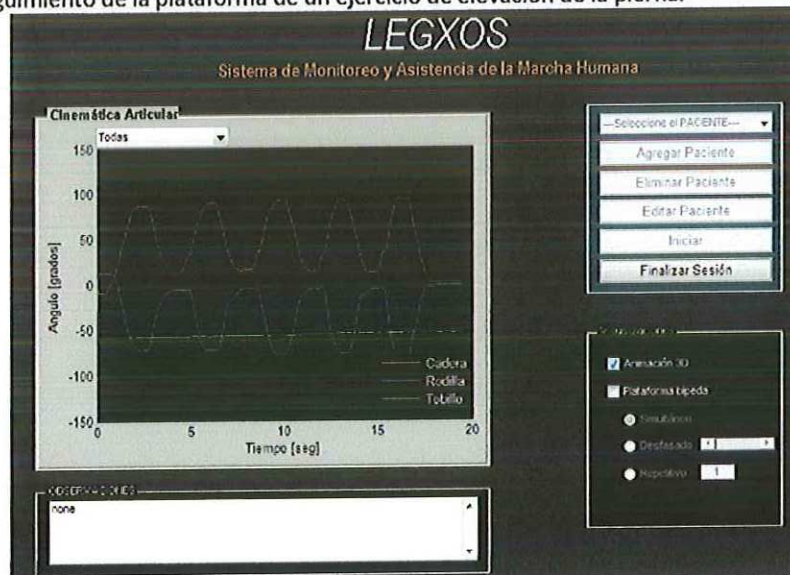
En la figura 62 se observa la señal adquirida y reproducida en los actuadores con una trayectoria de 5 movimientos de flexión y extensión de la cadera.

Figura 63. Seguimiento de un ejercicio de flexión y extensión de la rodilla



Fuente: El autor del proyecto.

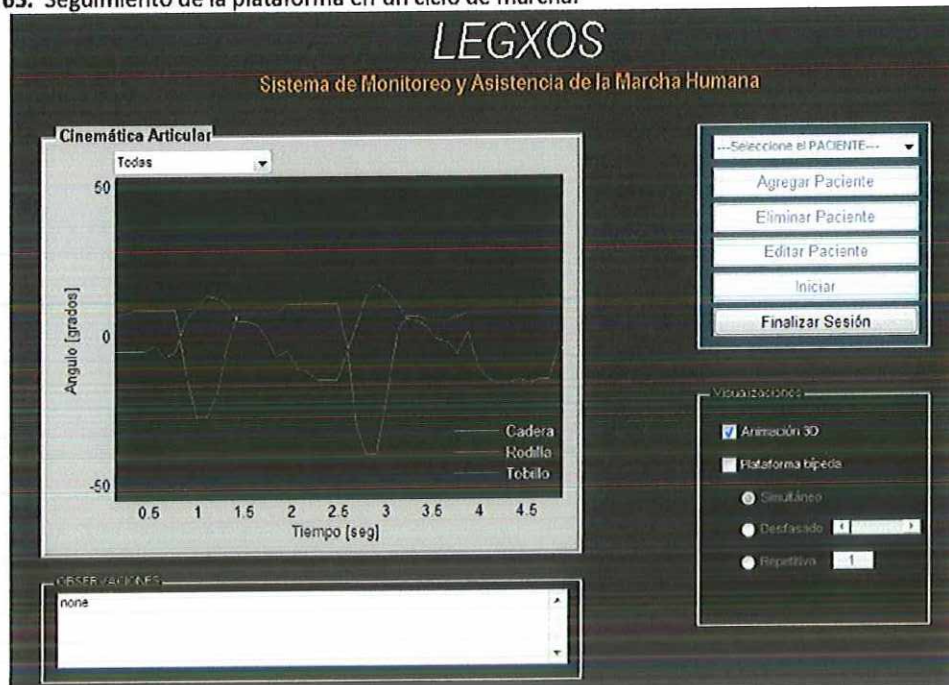
Figura 64. Seguimiento de la plataforma de un ejercicio de elevación de la pierna.



Fuente: El autor del proyecto.

En la figura 63 se observa cómo se sigue la trayectoria suave de la rodilla en una prueba de monitoreo de un ejercicio de flexión-extensión de 5 repeticiones, mientras que en la figura 64 se observa un ejercicio de levantamiento de pierna.

Figura 65. Seguimiento de la plataforma en un ciclo de marcha.



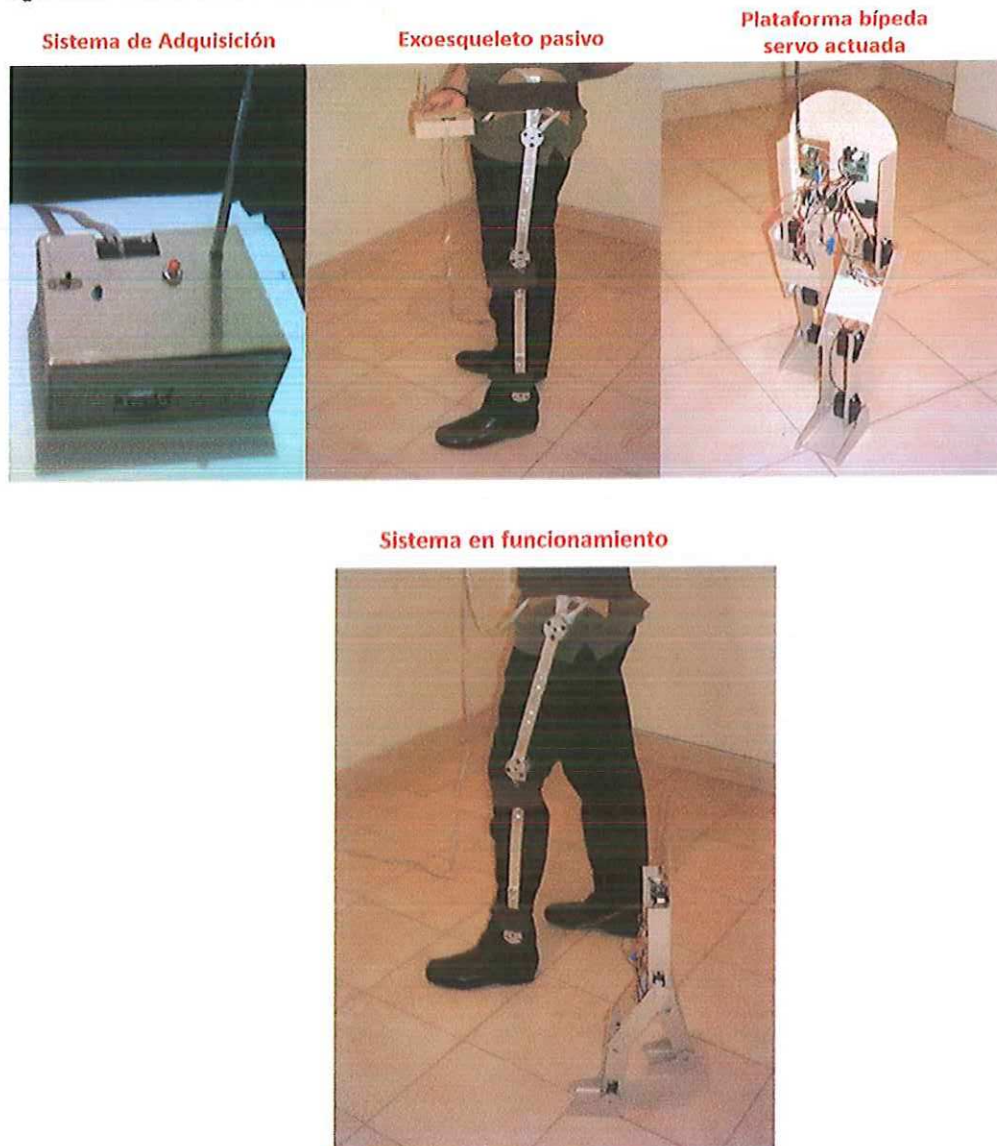
Fuente: El autor del proyecto.

En la figura 65 se puede observar un ciclo completo de la marcha. Si se compara con las curvas teóricas de la cinemática del miembro inferior descritas en el capítulo 2, se puede observar que tanto la rodilla como la cadera siguen una envolvente similar y los valores picos positivos y negativos coinciden alrededor de  $10^\circ$  y  $-45^\circ$  para la rodilla, y de  $20^\circ$  y  $-10^\circ$  para la cadera.

### Otras observaciones

Se realizaron diversas pruebas modulares en el software LabView, específicamente pruebas de comunicación serial y pruebas para creación de entornos virtuales, a fin de evaluar su estabilidad en los puntos de falla de la plataforma sobre la que se desarrolló la interfaz de este proyecto. Se encontró que para la comunicación serial LabView se comporta mucho mejor ya que es un programa especializado para la instrumentación. Aún así, esta plataforma de programación presenta ciertas limitantes en cuanto a la implementación de simulaciones con modelos gráficos de entornos virtuales en 3D y con la codificación de algoritmos matemáticos complejos. Con esta inquietud, y queriendo aprovechar las fortalezas de Matlab y LabView para concebir una aplicación más robusta, se está evaluando continuar desarrollando el alcance de este proyecto en una siguiente etapa, en la que se intercomunicaran las operaciones que se realicen en cada plataforma.

**Figura 66.** Resultados del modelo experimental.



Fuente: El autor del proyecto.

En la figura 66 se observan los resultados obtenidos del modelo experimental. En la parte superior de la figura encontramos la unidad de adquisición de señales, el exoesqueleto pasivo y la plataforma bípeda experimental. En la parte inferior se observa el modelo de pruebas emulando un ciclo de marcha ejecutado por la persona.

## 9. OBSERVACIONES Y CONCLUSIONES

En este proyecto se ha presentado la propuesta de diseño de un exoesqueleto de piernas de tipo maestro-esclavo, cuya validación se realizó de manera modular en sistemas auxiliares como la plataforma bípoda, debido a los altos costos que implicaría la construcción de un primer prototipo a escala real para implementación directa sobre las personas.

En el desarrollo de esta investigación se hizo uso de varias herramientas computacionales con el objeto de explotar la validación de modelos matemáticos y diseños de ingeniería a través de simulaciones en entornos que permiten observar una aproximación del problema en estudio. Este es uno de los aportes importantes de los resultados obtenidos, pues se integran áreas de la ingeniería mecatrónica que no se han desarrollado en la Universidad Autónoma de Bucaramanga, como lo son la realidad virtual y los dispositivos hápticos, que permiten interactuar a la persona con los modelos computacionales directamente.

Respecto al análisis biomecánico de la marcha humana, se puede concluir que el formalismo de Winter para la cinemática articular basada en ángulos absolutos, no es de gran aplicación en este proyecto, ya que el exoesqueleto pasivo de adquisición de señales registra ángulos relativos a sistemas de referencia locales ubicados en cada articulación, por ello el modelamiento se realizó de forma diferente. Aun así, las tablas de datos antropométricos propuestos por dicho autor, sí fueron de gran utilidad para determinar los parámetros que afectan la dinámica del movimiento.

En lo referente al modelamiento matemático del sistema, se encontraron varias aproximaciones desarrolladas en estudios previos por otros autores, pero muchos de ellos despreciaban en su modelo dinámico el tensor de inercia de los cuerpos por tratar a los segmentos como masas puntuales. Aun así, este trabajo desarrolla el modelo dinámico para una pierna por métodos matemáticos, recursivos y gráficos, en los cuales se consideró apropiado incluir los parámetros inerciales ya que en la marcha humana uno de los factores más importantes para el torque calculado en cada articulación es la contribución de la velocidad angular del cuerpo rígido

Aunque no fue implementado sobre una persona, es importante que el diseño de sistemas exoesqueléticos sea realizado en conjunto con profesionales del área médica que conozcan la kinesiólogía, las patologías y los correctos funcionamientos de las partes del cuerpo que se buscan imitar, ya que errores en el alineamiento de una junta real con una mecánica o la diferencia leve en longitudes de los segmentos, causarían problemas de

incomodidad sobre la persona que lleva puesto el dispositivo, o podría generarle un trauma mayor debido a fuerzas de reacción inducidas por errores geométricos.

El beneficio de esta propuesta es aprovechar los resultados médicos que aseguran que las terapias con repetición de movimientos permiten establecer más rápidamente las conexiones entre las neuronas que se encuentran afectadas. Este sistema permitiría automatizar y optimizar las terapias haciéndolas más intensivas, y desplazaría el papel del terapeuta hacia uno donde la principal tarea sea el correcto diagnóstico y análisis de la patología.

Respecto a la interfaz desarrollada, se puede concluir que los resultados obtenidos en el proceso de monitoreo de movimientos y accionamiento de motores fueron satisfactorios, sin embargo es necesario que el diseño gráfico de la misma sea un poco más amigable, por ello se plantea la opción de fusionar la versatilidad gráfica de LabView para el diseño de paneles de control, con la fortaleza numérica y de simulación de sistemas que permite realizar Simulink. Esta integración ya se ha empezado a desarrollar para mejorar el primer modelo funcional del sistema presentado en este trabajo. Es necesario también desarrollar un modelo de animación más completo que brinde mayor realidad al avatar.

El sistema propuesto es simplemente un primer paso dentro del área de la robótica de rehabilitación, la cual está tomando gran fortaleza con el avance de los sistemas computacionales. La principal dificultad encontrada para concebir el proyecto fue la componente de diseño mecánico debido a la falta de un asesor en el área de Ingeniería Mecánica o Diseño Industrial, ya que interactuar con máquinas o emular el movimiento corporal es realmente complicado al requerirse de estructuras ergonómicas, además, hoy en día las investigaciones apenas se orientan al desarrollo de actuadores ligeros y potentes. Por ello se validó el movimiento en una plataforma auxiliar con servomotores de pequeña escala, ya que el interés de este proyecto es el seguimiento de trayectorias asignadas por un ente maestro sobre un esclavo, y no el desarrollo de los actuadores como tal.

La tarjeta de adquisición y control desarrollada en este proyecto es un dispositivo sencillo de bajo costo cuyo uso se puede ampliar a otras aplicaciones de adquisición de movimientos. Aunque la principal aplicación que se persiguió en este proyecto fue la teleoperación, la adquisición de la cinemática de la pierna se podría ampliar a otras partes del cuerpo o a otros sistemas con enfoques diferentes que actualmente son bastantes comerciales, sobre todo en lo relacionado con animaciones de personajes computacionales, por ejemplo en sistemas educativos para cirugía en medicina, donde se interactúe con pacientes virtuales, o en la industria de los videojuegos o entretenimientos donde se podría generar el movimiento de los personajes con nuestro propio cuerpo.

## 10. Bibliografía

1. **BOWKER P MESSENGER N, OGILIVE C, ROWLEY DI** *Energetics of paraplegic walking*. 1993.
2. **CEPEDA, Rudy y OCHOA, Claudia** *Diseño de un equipo prototipo para la medición de ángulos articulares del miembro inferior en el plano sagital utilizando acelerómetros*. Universidad Industrial de Santander, Bucaramanga, 2006.
3. **G. LOEB F. RICHMON** *Neural prosthesis for restoration of sensor and motor function*. Boca Ratón, 2000.
4. **GUIZZO, Erico y GOLDSTEIN, Harry** *The rise of the Body Bots*. IEEE Spectrum, 2005.
5. **HOCOMA** [En línea]. - [http://www.hocoma.ch/web/en/products/introduction\\_lokomat.html](http://www.hocoma.ch/web/en/products/introduction_lokomat.html).
6. **MARSOLIS, Kobetic** *Functional walking in paralyzed patients, by means of electric stimulation*. 1983.
7. **PERRY, J.** *Gait Analysis: Normal and Pathological Function*. 1992.
8. **Sankai Yoshiyuki** CYBERDYNE [En línea]. - 12 de Junio de 2007. - <http://www.cyberdyne.jp/english/index.html>.
9. **SCHIELE, André y Van der HELM, Frans** *Kinematic Design to Improve Ergonomics in Human Machine Interaction*. IEEE TRANSACTIONS ON NEURAL SYSTEMS AND REHABILITATION ENGINEERING, 2006. - Vol. 14.
10. **SOTELANO, Fernando** *Estimulación Eléctrica Funcional (EEF): Estado actual en el hemipléjico adulto*. Revista Neurológica Argentina, 2003.
11. **TORRES F. [y otros]** *Robots y sistemas sensoriales*. - Madrid : Prentice Hall, 2002.
12. **VELE, Oscar.** *Diseño y construcción de un robot bípedo experimental*. Cuenca, Ecuador, Diciembre 2005.
13. **VILLANUEVA D. MUÑOZ, R., HERNANDEZ, PR.** *Técnicas de asistencia para la recuperación de la locomoción funcional después de una lesión de médula espinal*, 2001. - Vol. XXII : págs. 89-100.
14. **WESLEY, Eby.** *Feasibility Analysis of a Powered Lower-Limb Orthotic for the Mobility Impaired User*. Waterloo, 2005.
15. **WINTER, D.** *Biomechanics and motor control of human movement*. John Wiley & Sons. Toronto, 1979.

---

---

# **ANEXOS**

---

---



**ANEXO A.** Tablas de D. Winter para la cinemática de un ciclo de marcha.

TABLE A.4 Relative Joint Angular Kinematics—Ankle, Knee, and Hip

	FRAME	TIME	THETA	ANKLE	ALPHA	THETA	KNEE	ALPHA	THETA	HIP	ALPHA	
		S	DEG	OMEGA	R/S/S	DEG	OMEGA	R/S/S	DEG	OMEGA	R/S/S	
TOR	1	0.000	-15.2	-2.29	94.89	46.7	6.74	-21.91	-2.4	2.39	36.03	
	2	0.014	-16.4	-0.82	98.72	52.1	6.23	-46.59	-0.2	2.89	31.50	
	3	0.029	-16.5	0.54	84.63	56.9	5.41	-65.29	2.3	3.30	24.37	
	4	0.043	-15.6	1.60	63.13	61.0	4.37	-77.66	5.2	3.58	14.23	
	5	0.057	-13.9	2.34	41.31	64.1	3.19	-84.77	8.2	3.70	1.30	
	6	0.072	-11.7	2.78	20.10	66.2	1.94	-87.62	11.3	3.62	-12.28	
	7	0.086	-9.3	2.92	-0.74	67.3	0.68	-86.54	14.2	3.35	-23.42	
	8	0.100	-6.9	2.76	-18.70	67.3	-0.53	-81.99	16.8	2.95	-30.30	
	9	0.114	-4.8	2.38	-30.23	66.4	-1.66	-75.17	19.0	2.48	-33.55	
	10	0.129	-3.0	1.90	-34.13	64.6	-2.68	-67.64	20.8	1.99	-35.01	
	11	0.143	-1.7	1.41	-32.12	62.0	-3.60	-60.31	22.2	1.48	-35.53	
	12	0.157	-0.7	0.98	-26.84	58.7	-4.41	-53.09	23.3	0.98	-34.42	
	13	0.172	-0.1	0.64	-20.23	54.8	-5.11	-45.78	23.8	0.50	-30.99	
	14	0.186	0.3	0.40	-13.59	50.3	-5.72	-38.95	24.1	0.09	-26.06	
	15	0.200	0.6	0.25	-8.03	45.4	-6.23	-32.91	24.0	-0.25	-21.52	
	16	0.215	0.7	0.17	-3.87	40.1	-6.66	-26.57	23.7	-0.53	-18.38	
	17	0.229	0.9	0.14	-0.56	34.5	-6.99	-18.05	23.1	-0.77	-16.10	
	18	0.243	1.0	0.15	2.33	28.7	-7.17	-5.87	22.4	-0.99	-13.65	
	19	0.257	1.1	0.21	4.60	22.8	-7.16	10.65	21.5	-1.16	-10.37	
	20	0.272	1.3	0.29	5.47	16.9	-6.87	31.09	20.5	-1.28	-6.16	
	21	0.286	1.6	0.36	3.83	11.5	-6.27	53.59	19.4	-1.34	-1.66	
	22	0.300	1.9	0.40	-0.84	6.7	-5.34	74.76	18.3	-1.33	1.98	
	23	0.315	2.2	0.34	-8.36	2.7	-4.13	90.62	17.2	-1.28	4.14	
	24	0.329	2.5	0.16	-19.22	-0.1	-2.75	97.96	16.2	-1.21	5.38	
	25	0.343	2.5	-0.21	-33.28	-1.8	-1.33	95.28	15.2	-1.13	6.53	
	26	0.357	2.1	-0.79	-46.39	-2.3	-0.02	82.64	14.3	-1.03	7.45	
	27	0.372	1.2	-1.54	-50.65	-1.8	1.04	62.40	13.6	-0.92	7.62	
	HCR	28	0.386	-0.4	-2.24	-38.30	-0.6	1.76	40.55	12.8	-0.81	7.93
		29	0.400	-2.5	-2.63	-7.99	1.1	2.20	24.37	12.2	-0.69	9.91
		30	0.415	-4.7	-2.47	30.12	3.0	2.46	16.62	11.7	-0.52	13.66
		31	0.429	-6.5	-1.77	59.60	5.1	2.67	12.93	11.4	-0.30	16.99
		32	0.443	-7.6	-0.77	70.70	7.4	2.83	5.97	11.2	-0.04	16.09
33		0.458	-7.8	0.25	64.54	9.8	2.84	-8.17	11.3	0.16	7.69	
34		0.472	-7.2	1.08	46.87	12.1	2.60	-27.06	11.5	0.18	-7.32	
35		0.486	-6.0	1.59	24.33	14.0	2.07	-44.04	11.6	-0.05	-22.88	
36		0.500	-4.6	1.77	4.14	15.4	1.34	-53.13	11.4	-0.47	-32.00	
37		0.515	-3.1	1.71	-8.73	16.2	0.55	-52.57	10.8	-0.96	-31.67	
38		0.529	-1.8	1.52	-13.85	16.3	-0.17	-44.30	9.8	-1.38	-23.59	
39		0.543	-0.6	1.31	-13.32	15.9	-0.72	-31.51	8.6	-1.64	-11.54	
40		0.558	0.4	1.14	-9.71	15.2	-1.07	-17.12	7.2	-1.71	0.90	
41		0.572	1.2	1.04	-6.42	14.2	-1.21	-3.98	5.8	-1.61	10.98	
42		0.586	2.1	0.96	-6.79	13.2	-1.18	5.07	4.5	-1.40	16.74	
43		0.601	2.8	0.84	-10.34	12.3	-1.06	8.50	3.5	-1.13	17.58	
44		0.615	3.4	0.66	-12.36	11.4	-0.94	7.49	2.7	-0.89	15.03	
45		0.629	3.9	0.49	-9.52	10.7	-0.85	4.99	2.0	-0.70	11.60	
46		0.643	4.2	0.39	-3.26	10.1	-0.80	3.03	1.5	-0.56	8.39	
47		0.658	4.6	0.40	3.33	9.4	-0.76	1.90	1.1	-0.46	4.74	
48		0.672	4.9	0.49	7.14	8.8	-0.74	1.14	0.8	-0.43	-0.13	
49		0.686	5.3	0.60	4.92	8.2	-0.73	0.45	0.4	-0.47	-5.26	
50		0.701	5.9	0.63	-3.15	7.6	-0.73	0.18	0.0	-0.58	-8.17	
51		0.715	6.4	0.51	-11.50	7.0	-0.72	1.21	-0.5	-0.70	-7.39	
52		0.729	6.7	0.30	-14.34	6.4	-0.69	4.06	-1.2	-0.79	-4.00	
53		0.744	6.9	0.10	-10.68	5.9	-0.61	8.77	-1.8	-0.81	-0.17	
54		0.758	6.9	-0.01	-4.17	5.4	-0.44	15.06	-2.5	-0.79	3.12	
55		0.772	6.9	-0.02	0.00	5.2	-0.18	22.07	-3.1	-0.73	5.70	
56		0.786	6.8	-0.01	-1.14	5.1	0.19	28.41	-3.7	-0.63	6.92	
57		0.801	6.8	-0.05	-7.42	5.5	0.63	32.98	-4.2	-0.53	6.16	
58		0.815	6.8	-0.22	-17.46	6.2	1.13	35.54	-4.6	-0.45	3.95	
59		0.829	6.5	-0.55	-28.75	7.3	1.65	36.64	-4.9	-0.41	1.81	
60		0.844	5.9	-1.04	-37.44	8.9	2.18	37.19	-5.2	-0.40	1.24	
61		0.858	4.8	-1.62	-40.93	10.9	2.71	37.90	-5.6	-0.38	2.74	
62		0.872	3.2	-2.21	-40.94	13.3	3.26	39.00	-5.9	-0.32	5.62	
63		0.887	1.2	-2.79	-41.67	16.2	3.83	40.04	-6.1	-0.22	8.61	
64		0.901	-1.4	-3.40	-43.21	19.6	4.41	39.95	-6.2	-0.08	11.23	
65		0.915	-4.4	-4.03	-38.18	23.5	4.97	37.59	-6.2	0.10	14.31	
66		0.929	-8.0	-4.50	-17.72	27.8	5.48	32.25	-6.0	0.33	18.85	
67		0.944	-11.8	-4.54	18.41	32.4	5.90	23.44	-5.7	0.64	24.81	
68		0.958	-15.4	-3.97	59.42	37.4	6.15	10.87	-5.0	1.04	31.17	
69		0.972	-18.3	-2.84	91.15	42.5	6.21	-4.80	-4.0	1.53	36.24	

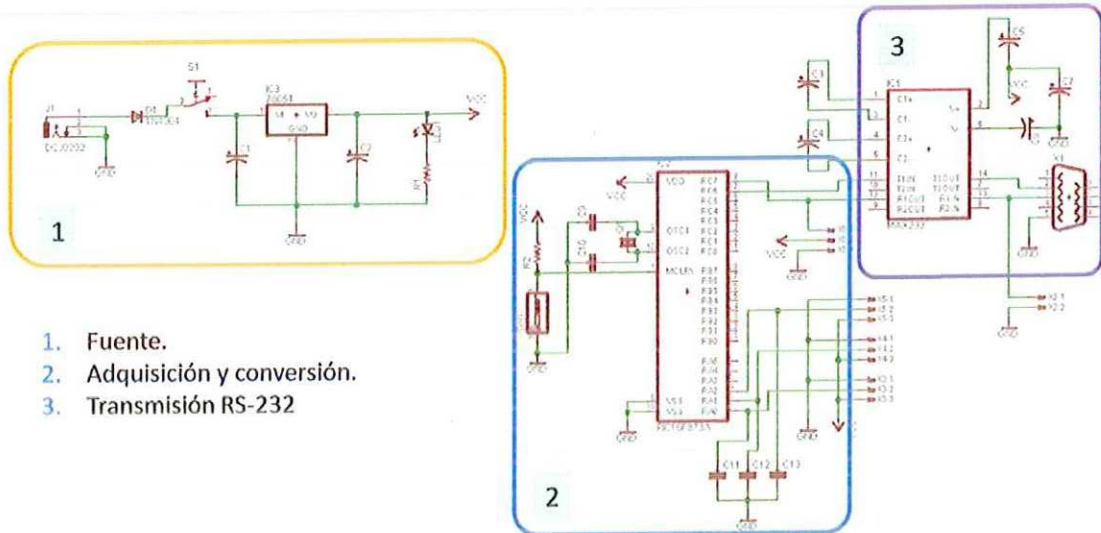
TABLE A.4 (Continued)

	FRAME	TIME	THETA	ANKLE	ALPHA	THETA	KNEE	ALPHA	THETA	HIP	ALPHA
		S	DEG	OMEGA	R/S/S	DEG	OMEGA	R/S/S	DEG	OMEGA	R/S/S
TOR	70	0.987	-20.1	-1.36	104.21	47.6	6.02	-21.92	-2.5	2.08	38.01
	71	1.001	-20.5	0.14	96.45	52.4	5.58	-38.75	-0.5	2.62	34.78
	72	1.015	-19.8	1.40	72.74	56.7	4.91	-53.72	1.8	3.07	26.42
	73	1.030	-18.2	2.22	42.50	60.4	4.04	-65.55	4.5	3.38	14.66
	74	1.044	-16.2	2.61	15.01	63.4	3.03	-74.32	7.3	3.49	1.29
	75	1.058	-14.0	2.65	-3.93	65.4	1.92	-80.71	10.2	3.41	-12.63
	76	1.072	-11.8	2.50	-13.24	66.5	0.73	-83.96	12.9	3.13	-25.25
	77	1.087	-9.9	2.28	-16.29	66.6	-0.48	-82.70	15.3	2.69	-33.71
	78	1.101	-8.1	2.03	-17.51	65.7	-1.64	-76.99	17.3	2.17	-36.16
	79	1.115	-6.5	1.77	-18.73	63.9	-2.69	-68.11	18.9	1.66	-32.81
	80	1.130	-5.2	1.50	-19.65	61.3	-3.59	-57.74	20.1	1.23	-25.82
	81	1.144	-4.1	1.21	-19.37	58.0	-4.34	-48.02	20.9	0.92	-18.62
	82	1.158	-3.2	0.94	-17.21	54.2	-4.96	-40.91	21.6	0.70	-14.21
	83	1.173	-2.5	0.72	-13.87	49.9	-5.51	-37.09	22.1	0.51	-13.55
	84	1.187	-2.0	0.55	-10.79	45.2	-6.02	-35.35	22.4	0.31	-15.62
	85	1.201	-1.6	0.41	-7.88	40.0	-6.52	-32.90	22.6	0.06	-18.75
	86	1.215	-1.3	0.32	-3.38	34.5	-6.96	-26.57	22.5	-0.23	-21.73
	87	1.230	-1.1	0.31	3.27	28.6	-7.28	-14.15	22.2	-0.56	-23.63
	88	1.244	-0.8	0.42	9.14	22.6	-7.37	4.76	21.6	-0.90	-23.22
	89	1.258	-0.4	0.58	10.06	16.5	-7.14	28.98	20.7	-1.22	-19.01
	90	1.273	0.1	0.70	4.80	10.9	-6.54	56.01	19.6	-1.45	-10.61
	91	1.287	0.7	0.71	-4.67	5.8	-5.54	81.46	18.3	-1.52	-0.51
	92	1.301	1.3	0.57	-16.27	1.8	-4.21	99.89	17.1	-1.46	6.98
	93	1.316	1.7	0.25	-29.67	-1.1	-2.68	107.07	15.9	-1.32	9.49
	94	1.330	1.7	-0.28	-44.20	-2.6	-1.15	101.58	14.9	-1.19	8.22
	95	1.344	1.2	-1.02	-54.99	-3.0	0.22	85.39	14.0	-1.09	6.14
	96	1.358	0.0	-1.85	-53.10	-2.3	1.30	63.88	13.1	-1.01	5.89
HCR	97	1.373	-1.8	-2.53	-31.93	-0.8	2.05	43.96	12.3	-0.92	8.35
	98	1.387	-4.1	-2.77	4.96	1.1	2.55	29.90	11.6	-0.78	12.18
	99	1.401	-6.4	-2.39	43.33	3.4	2.90	20.18	11.1	-0.57	14.65
	100	1.416	-8.0	-1.53	67.62	5.9	3.13	9.21	10.7	-0.36	13.20
	101	1.430	-8.9	-0.46	71.62	8.5	3.17	-6.47	10.5	-0.20	6.83
	102	1.444	-8.8	0.52	59.88	11.0	2.94	-24.49	10.4	-0.16	-2.99
	103	1.459	-8.0	1.25	41.13	13.3	2.47	-39.25	10.2	-0.28	-12.86
	104	1.473	-6.7	1.70	22.10	15.1	1.82	-46.90	9.9	-0.53	-19.17
	105	1.487	-5.2	1.89	5.23	16.3	1.13	-47.03	9.4	-0.83	-19.53
	106	1.501	-3.7	1.85	-10.10	16.9	0.48	-40.67	8.5	-1.09	-13.02

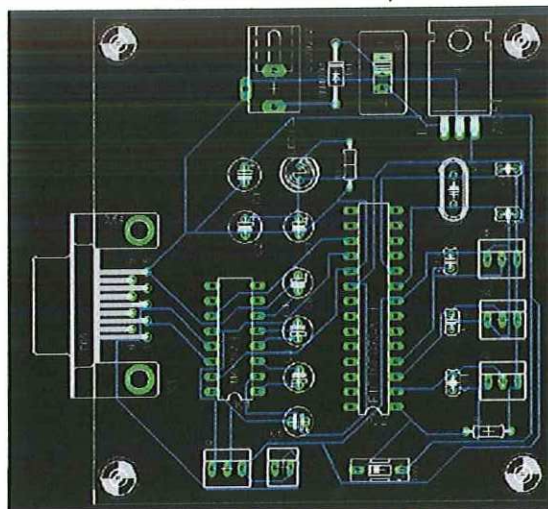
## ANEXO B. Esquemáticos

### Unidad local del procesamiento

#### Tarjeta de adquisición de datos y conversión A/D



#### Diseño del Circuito Impreso





## Features

- Conductive plastic
- Linear tapers
- Audio tapers available
- PC board and bushing mount
- Plastic bushing and plastic shaft

- Withstands typical industrial washing processes
- RoHS compliant\* version available

## 3360 - 6 mm Square Sealed Panel Control

### Electrical Characteristics

Standard Resistance Range  
 Linear.....1K ohms to 1 megohm  
 Total Resistance Tolerance  
 Linear Tapers .....±20 %  
 Independent Linearity.....±5 %  
 Absolute Minimum Resistance  
 .....2 % or 2 ohms max.  
 (whichever is greater)  
 Effective Electrical Angle .....240 ° nominal  
 Contact Resistance Variation  
 .....1 % or 3 ohms max.  
 (whichever is greater)  
 Dielectric Withstanding Voltage  
 (MIL-STD-202 - Method 301)  
 Sea Level.....900 VAC minimum  
 70,000 Feet .....350 VAC minimum  
 Insulation Resistance  
 .....1,000 megohms minimum  
 Power Rating @ 70 °C  
 (Derate to 0 at 125 °C)  
 Voltage Limited By Power Dissipation  
 or 200 VAC, Whichever is Less)  
 .....25 watts  
 Audio Tapers.....0.125 watts  
 Theoretical Resolution....Essentially infinite

### Environmental Characteristics

Operating Temperature Range  
 .....+1 °C to +125 °C  
 Storage Temperature Range  
 .....-55 °C to +125 °C  
 Temperature Coefficient  
 Over Storage Temperature Range  
 ..... ±1,000 ppm/°C  
 Vibration .....30 G  
 Total Resistance Shift .... ±2 % maximum  
 Voltage Ratio Shift ..... ±2 % maximum  
 Shock .....100 G  
 Total Resistance Shift ....±2 % maximum  
 Voltage Ratio Shift .....±2 % maximum  
 Load Life.....1,000 Hours  
 Total Resistance Shift  
 .....±10 % TRS maximum  
 Rotational Life (No Load) .....50,000 Cycles  
 Total Resistance Shift  
 .....±5 % TRS maximum  
 Contact Resistance Variation  
 .....3 % or 3 ohms, whichever is greater  
 Moisture Resistance  
 MIL-STD-303, Method 103, Condition B  
 Total Resistance Shift  
 .....±10 % TRS maximum  
 IP Rating .....IP 67

### Physical Characteristics

Stop Strength.....5.65 N-cm (8 oz.-in.)  
 Mechanical Angle .....310 ° nominal  
 Torque  
 Starting  
 .....3.53 N-cm (5.0 oz.-in.) maximum  
 Running  
 .....3.53 N-cm (5.0 oz.-in.) maximum  
 Mounting (Torque on Bushing)  
 .....17 N-cm (1.5 lb.-in.) maximum  
 [plastic bushing]  
 Weight .....4.5 grams  
 Terminals .....Solderable pins  
 Soldering Condition  
 .....5 seconds at 360 °C maximum.  
 Recommended hand soldering  
 using Sn95/Ag5 no clean solder,  
 0.025 wire diameter.  
 Marking .....Manufacturer's trademark,  
 model number, product code,  
 terminal style, resistance code  
 and date code  
 Flammability.....Conforms to UL94V-0  
 Epoxy.....Conforms to UL94V-1  
 Hardware (purchased separately)  
 Nut.....H-38-1  
 Washer .....H-37-5

### Standard Resistance Table

Resistance (Ohms)	Resistance Code
<b>1,000</b>	<b>102</b>
<b>2,000</b>	<b>202</b>
<b>5,000</b>	<b>502</b>
<b>10,000</b>	<b>103</b>
<b>20,000</b>	<b>203</b>
<b>50,000</b>	<b>503</b>
<b>100,000</b>	<b>104</b>
<b>200,000</b>	<b>204</b>
<b>500,000</b>	<b>504</b>
<b>1,000,000</b>	<b>105</b>

Popular distribution resistance values listed in boldface. Special resistances available.

### How To Order

**3360 Y - 1 - 103 LF**

Model \_\_\_\_\_  
 Terminal Style Designator \_\_\_\_\_  
 Single Cup: \_\_\_\_\_  
 C = In-line Straight  
       Terminals Side Exit  
 P = 5.08 mm x 2.54 mm  
       Triangular Pattern  
       Rear Exit  
 Y = 5.08 mm x 5.08 mm  
       Triangular Pattern  
       Rear Exit  
 Bushing Designator \_\_\_\_\_  
 1 = Standard  
 Resistance Code \_\_\_\_\_  
 Terminations \_\_\_\_\_  
 LF = 100 % Tin-plated (RoHS compliant)  
 Blank = 90 % Tin / 10 % Lead-plated  
 (Standard)



Reliable Electronic Solutions

**Asia-Pacific:**  
 TEL +886-2 25624117  
 FAX +886-2 25624116

**Europe:**  
 TEL +41-41 7685555  
 FAX +41-41 7685510

**The Americas:**  
 TEL +1-951 781-5500  
 FAX +1-951 781-5700

[www.bourns.com](http://www.bourns.com)

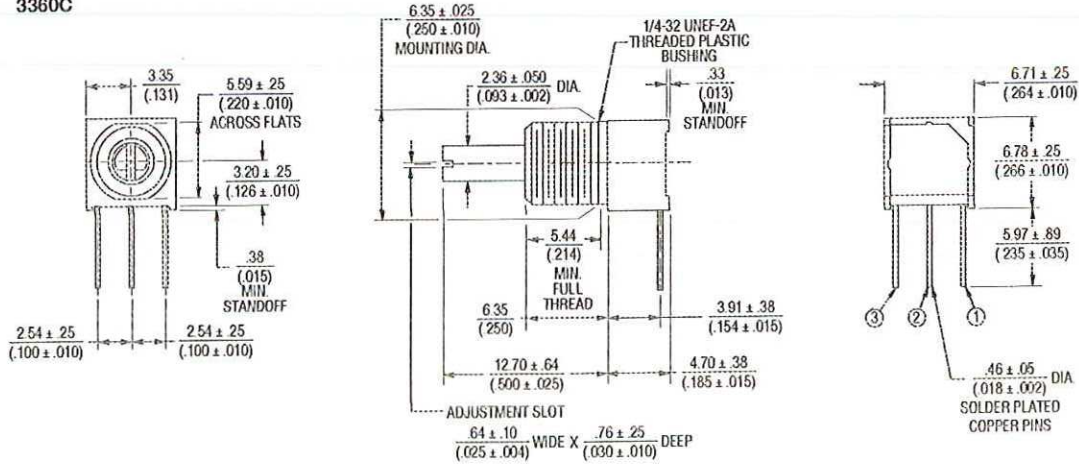
\*RoHS Directive 2002/95/EC Jan 27 2003 including Annex.  
 Specifications are subject to change without notice.  
 Customers should verify actual device performance in their specific applications.

# 3360 - 6 mm Square Sealed Panel Control

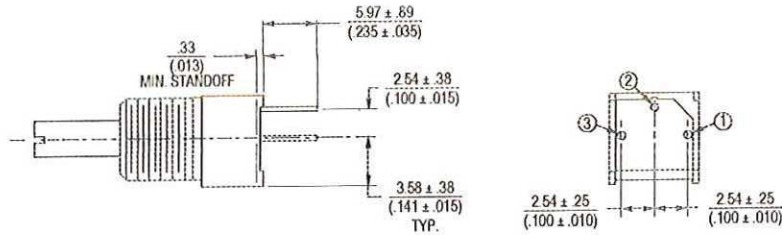
**BOURNS®**

## Product Dimensions

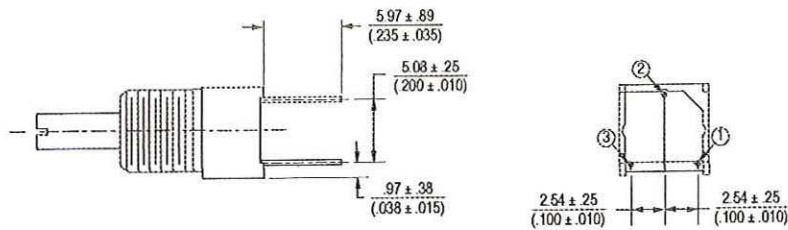
### COMMON DIMENSIONS 3360C



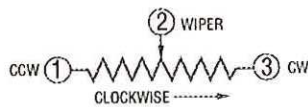
### 3360P



### 3360Y



DIMENSIONS ARE:  $\frac{\text{MM}}{\text{(INCHES)}}$



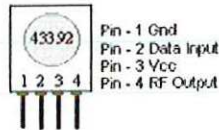
REV. 02/06

Specifications are subject to change without notice.  
Customers should verify actual device performance in their specific applications.

## ANEXO C. - Especificaciones de los Módulos RF

TWS-434 / RWS-434  
<http://www.rentron.com>

### TWS-434A RF Transmitter

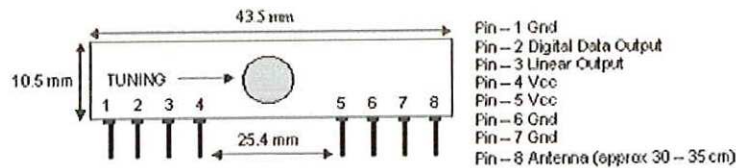


**Frequency:** 433.92MHz  
**Modulation:** AM  
**Operating Voltage:** 2 - 12 VDC

Module size W = 0.418" H = 0.480" lead spacing 0.1"

Symbol	Parameter	Condition	Min	Typ	Max	Unit
Vcc	Supply Voltage		2.0	-	12.0	V
I <sub>p</sub>	Peak Current	2V / 12V	-	1.64 / 19.4	-	mA
V <sub>h</sub>	Input High Voltage	I <sub>data</sub> = 100uA (High)	Vcc-0.5	Vcc	Vcc+0.5	V
V <sub>l</sub>	Input Low Voltage	I <sub>data</sub> = 0 uA (Low)	-	-	0.3	V
F <sub>o</sub>	Operating Frequency		433.90	433.92	433.94	MHz
T <sub>r</sub> / T <sub>f</sub>	Modulation Rise / Fall Time	External Coding	-	-	100 / 100	uS
P <sub>o</sub>	RF Output Power - Into 50Ω	Vcc = 9 to 12 V Vcc = 5 to 6V	-	16 14	-	dBm
D <sub>r</sub>	Data Rate	External Coding	-	2.4K	3K	Bps

### RWS-434 RF Receiver



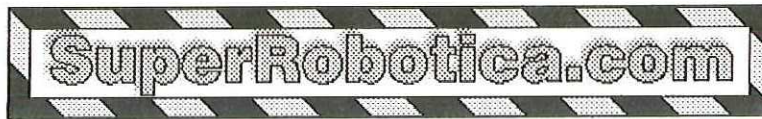
**Frequency:** 433.92MHz  
**Modulation:** AM  
**Operating Voltage:** 4.5 - 5.5 VDC  
**Output:** Digital & Linear

Symbol	Parameter	Condition	Min	Typ	Max	Unit
Vcc	Supply Voltage		4.5	5	5.5	V
I <sub>t</sub>	Operating Current		-	3.5	4.5	mA
	Channel Width	+ / - 500				kHz
R <sub>d</sub>	Data Rate				3k	Bps
V <sub>dat</sub>	Data Out	I <sub>data</sub> = +200 uA (High)	Vcc-0.5	-	Vcc	V
		I <sub>data</sub> = -10 uA (Low)	-	-	0.3	V

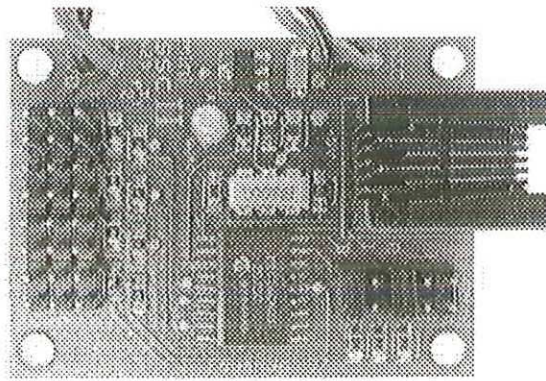
Reynolds Electronics  
 3101 Eastridge Lane  
 Canon City, CO. 81212  
 Tel: (719) 269-3469 Fax: (719) 276-2853  
 Web: <http://www.rentron.com>

**ANEXO D. Especificaciones y operación de la tarjeta controladora de servos**





**Manual de instrucciones**  
**CONTROLADOR 8 SERVOS MINI SSC S310165**



Traducción española por Alicia Bernal.

## Índice de contenidos

Conectores y jumpers de configuración	2
Cómo configurar el controlador Mini SSC II	2
Cómo conectar el controlador SSC II	3
Comprobaciones iniciales	5
Cómo programar el controlador Mini SSC II	5
Uso de varios controladores Mini SSCs	5
Aspectos teóricos del funcionamiento de los servos	6
Ejemplos de programación	7

## Descargo de responsabilidades

Scott Edwards Electronics, Inc. no se hace responsable de ningún daño indirecto importante resultante de cualquier violación de la garantía, bajo cualquier jurisdicción incluyendo la pérdida de beneficios, tiempos de inactividad, gastos de reparaciones o imperfectos o sustitución de materiales. Así mismo, la violación de la garantía no cubre los gastos derivados de la recuperación, reprogramación y reproducción de los datos asociados con el uso del hardware o software descrito en este documento.

## Garantía

Scott Edwards Electronics, Inc. Garantiza este producto contra cualquier defecto de los materiales y funcionamiento durante un periodo de 90 días. Si detecta cualquier defecto, por nuestra parte optaremos por reparar, reemplazar o rembolsar el precio de compra. Para ello, envíenos una nota describiendo el problema y según corresponda procederemos a reemplazar el producto o rembolsar su valor a través de agencia de transporte. El servicio de transporte urgente correrá a cargo del cliente.

- Nota: la manipulación de manera inapropiada del módulo, así como la reparación por parte del usuario se considerará como una violación de la garantía.

## Marcas registradas y Copyrights

Mini SSC™ es una marca registrada de Scott Edwards Electronics, Inc.; BASIC Stamp® es una marca registrada de Parallax Inc. Todos los nombres de marcas registradas citados en este documento son propiedad de sus respectivos propietarios.

Este manual en su totalidad es propiedad de Scott Edwards Electronics, Inc., copyright© 1999.

## Conectores y jumpers de configuración

La figura 1 muestra la distribución de la placa de circuito del controlador de servos Mini SSC II con la posición de los conectores y los jumpers de configuración.

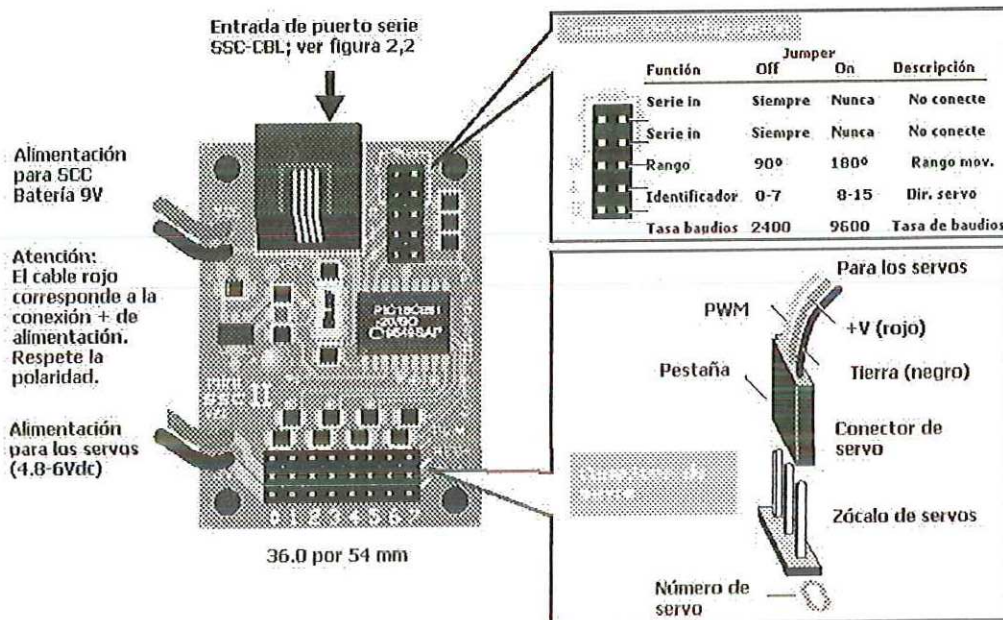


Figura 1. Diseño de la placa de circuito de Mini SSC II.

### iiii Atención!!!!

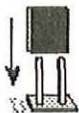
- \* No invierta la polaridad de la alimentación, ni siquiera de manera temporal, ya que podría destruir los componentes electrónicos.
- \* Utilice el voltaje máximo de 10 voltios en el terminal de entrada SCC 9V, ya que podría dañar definitivamente la unidad o acortar la vida del mismo.
- \* El conector jack se puede utilizar sólo como entrada serie. Conectar el jack a la línea telefónica podría dañar la unidad.

### Cómo configurar el controlador de servos Mini SSC II

La configuración por defecto del controlador Mini SSC II' es la siguiente (ningún jumper conectado):

Velocidad: 2.400 baudios • Servos 0 a 7 • Rango de movimiento = 90°

Para modificar la configuración, instale un jumpers en los pines adecuados como se indica en la ilustración siguiente. Los cambios serán efectivos la próxima vez que active el controlador SCC.



A continuación, encontrará los detalles de las opciones de la configuración:

**(R)ango:** Sin jumper en R, el controlador Mini SSC II controla servos con un rango de movimiento del 90°. Las posiciones de los servos están expresadas en unidades dentro del rango 0 -254, por lo que cada unidad corresponde con un cambio de 0.36° en la posición de los servos. Con un jumper en R, el controlador Mini SSC II controla servos más allá de los 180°, cada unidad corresponde con un cambio de posición de 0.72°.

**Nota acerca del rango de movimiento:**

Algunos servos no pueden moverse con un ángulo de 180°. Estos podrían atascarse si se utilizan posiciones inferiores a 50 o superiores a 200 con el controlador Mini SSC II en modo 180°. Le recomendamos que compruebe el funcionamiento de servos nuevos o distintos a los habituales antes de utilizarlos con el modo 180° del controlador Mini SSC II. Mueva el servo unas cuantas unidades de una vez hacia el final de su trayecto de movimiento para comprobar si se bloquea. Si lo hiciera, deberá utilizar el servo únicamente en modo de 90°, o restringir los valores de posición para definir un rango de movimiento seguro.

Debe tener en cuenta que los servos han sido diseñados para un rango de movimiento de 90°. El modo 180° se aprovecha de este margen de movimiento adicional pensado para las tolerancias mecánicas y eléctricas. En otras palabras, obtener un rango de 180° es como hacer trampas. No hay nada erróneo en un servo que no pueda ofrecerle un rango de 180°.

**(I)dentificación:** Sin jumper en I, las direcciones de los servos coincidirán como los números impresos en los zócalos de los servos—0 a 7. Con un jumper I, el controlador Mini SSC suma 8 direcciones, por lo que el servo conectado a "0" utilizará la dirección 8, conectado a "1" utilizará la 9 ... y conectado a "7" utilizará la dirección 15. Esta opción le permitirá conectar dos controladores Mini SSC II al mismo puerto serie y controlar los servos de manera individual. Para ello deberá consultar la información relativa al control de varios controladores Mini SSC desde un mismo puerto serie.

**(B)audios:** Sin jumper en B, el controlador Mini SSC II recibe los datos a través del puerto serie a una tasa de 2400 baudios; con un jumper en B, la tasa de baudios es 9600. En cualquier caso, los datos deberían enviarse como 8 bits de datos, sin paridad, 1 (o más) bit(s) de parada; N81 abreviado. Además, se deberían invertir los datos —del mismo modo que vienen de uno estándar

## Cómo conectar el controlador de servos Mini SSC II

**Servos:** el conector del Mini SSC II admite clavijas de servos de tres conductores (por ejemplo, los conectores J de Futaba o Hitec). Consulte la ilustración 1 para conocer la forma correcta de acoplar estos conectores.

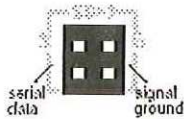
**Alimentación de servos:** Conecte el alimentador de los servos (4.8 a 6 VDC) a los cables rojos (+) y negros (-) marcados como SVO en la placa de circuitos del Mini SSC II. No invierta la polaridad de la alimentación, ni siquiera de manera momentánea, ya que podría dañar los servos. Como fuente de alimentación, utilice 4 pilas alcalinas tipo C o D. Para una alimentación AC, utilice un alimentador regulado lineal (no conmutado) de 5Vdc a 1A (o superior). Si desea obtener más información sobre la alimentación, visite [www.seetron.com/ssc\\_faq.htm](http://www.seetron.com/ssc_faq.htm).

**Alimentación del controlador Mini SSC II:** Conecte una pila de 9V al pack de pilas. Si desea utilizar otra fuente de alimentación, corte el cable del pack de pilas y conecte de 7 a 15Vdc a los cables, + al rojo y - al negro.

**No invierta la polaridad + y -, ni siquiera de manera momentánea.** Si lo hiciera podría dañar los componentes electrónicos. La entrada de alimentación del SSC está protegida contra cualquier inversión accidental de la batería de 9V. No obstante un voltaje de entrada superior podría deshabilitar esta protección.

**Entrada por puerto serie:** El controlador de servos Mini SSC II requiere sólo dos conexiones con una placa de circuito impreso—conexión de datos por puerto serie y la señal de tierra. Hay dos lugares en los que hacer estas conexiones; un conector "jack" de teléfono y un par de pines marcados como S(in) en el zócalo de configuración. No importa cual de ellos utilice—seleccione la conexión más cómoda.

Con microcontroladores de una sola placa como BASIC Stamp® probablemente utilizará los pines del zócalo. Conecte el pin I/O de Stamp que utilizará su programa para la salida por puerto serie a S en el controlador de servos Mini SSC II. En Stamp I se tratará de los pines 0—7, y en el BS2 se utilizarán los pines P0—P7. (No utilice el pin BS2 pin Sout; Sirve para comunicarse con su PC durante la programación.) En cualquier caso, conecte el pin de tierra del Stamp (Vss) a G.



Tenga en cuenta que hay dos pines S y dos pines G. Deberá realizar puntos de conexión adecuados para conectar varios controladores de servos Mini SSC II a la misma entrada serie. Consulte la sección que trata acerca de la conexión de varios controladores Mini SSC.

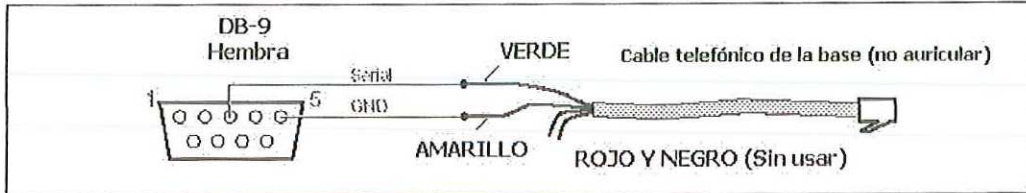


Figura 2. Conexión de un conector DB9 de puerto serie a un cable telefónico para utilizarlo con un SSC.

La figura 2 anterior le indica cómo debe conectar el cable SSC para utilizarlo junto con un conector DB9 de la placa de circuito impreso y otros ordenadores. Si prefiere no fabricar su propio cable, podrá adquirir un cable similar al SSC-CBL en cualquier tienda especializada. En la figura 3 encontrará el método de fabricación de este cable:

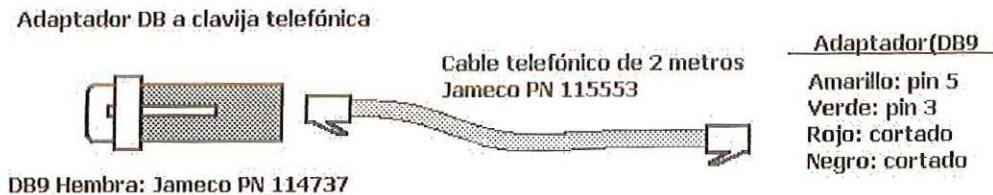


Figura 3. Cable SSC sin soldadura (equivalente a SSC-CBL)

El adaptador DB9-a-modular viene sin montar. Para montarlo, introduzca los terminales en las clavijas numeradas tal y como se indica en la imagen anterior. Introduzca el cable telefónico en la carcasa del adaptador. Tenga en cuenta que puede utilizar cualquier cable telefónico diseñado para la conexión con la clavija telefónica de pared. El único requisito es que el cable sea de cuatro hilos (con los cables amarillo, verde, rojo y negros todos conectados). Los cables de dos hilos que normalmente traen los teléfonos más baratos tienen sólo los cables rojo y negro. Estos cables no funcionarán. Puede comprobar qué tipo de cable es mirando a través de la clavija de plástico transparente.

**!!!!Nunca conecte el controlador Mini SSC II a la línea telefónica!!!!**

**Puertos serie de 25 pines:** Estos puertos ya casi no se utilizan, pero si su ordenador dispone de un puerto de este tipo, deberá utilizar un adaptador estándar de DB25-a-DB9 y uno de los cables descritos anteriormente.

### **Comprobaciones iniciales**

Cuando conecte por primera vez el controlador de servos Mini SSC II, se iluminará el LED Sync y se moverán todos los servos a la posición central (consulte la sección Aspectos teóricos del funcionamiento de los servos). Una vez que haya conectado los servos, la alimentación de los mismos y del controlador Mini SSC, los servos se moverán inmediatamente a su posición central. Si ya se encuentran en dicha posición central, entonces no se moverán mucho, pero puede comprobar el funcionamiento correcto del controlador moviendo suavemente con los dedos el servo lejos de la posición central. El servo debería intentar resistir el movimiento. En caso contrario, vuelva a comprobar las conexiones de los servos y la alimentación.

**Indicador verde:** Al encender el sistema, el LED Sync de color verde indica no sólo que controlador Mini SSC II está recibiendo alimentación, sino que su microcontrolador está funcionando correctamente (ha pasado la iniciación de arranque). Si se enciende la luz verde, pero los servos no parecen estar respondiendo correctamente, es probable que haya algún error en las conexiones de los servos, la alimentación de los mismos, o conexiones del puerto serie, y NO en el controlador Mini SSC II propiamente dicho.

**Comprobación del puerto serie:** Para verificar que se han establecido correctamente las conexiones del puerto serie, ejecute uno de los programas demo, un paquete de software del Mini SSC II como el visual SC disponible en <http://www.superrobotica.com/VisualSC.htm> o un programa de prueba para puertos serie. Compruebe que la tasa de baudios del programa coincide con la configuración del Mini SSC II, y que el pin I/O o puerto comm seleccionado para la salida serie corresponde con el puerto al que está conectado el controlador de servos Mini SSC II.

### **Cómo programar el controlador de servos Mini SSC II**

Para ordenar a un servo que se coloque en una determinada posición es necesario enviar tres bytes con la tasa de baudios adecuada (2400 o 9600 baudios, dependiendo de la configuración del jumper B; consulte el apartado Cómo configurar el controlador de servos Mini SSC II). Estos bytes:

Byte 1	Byte 2	Byte 3
[marcador de sincronismo (255)]	[Nº de servo (0-254)]	[posición (0-254)]

Estos bytes deben ser enviados como valores de bytes individuales, no como representaciones de texto de los números que podría escribir en un teclado. Los programas de ejemplo que se incluyen al final de este manual le indican cómo puede convertir números a valores de bytes. En PBASIC, simplemente omite las funciones de formato de texto (# para BS1; DEC, HEX, ?, etc. para BS2). En otros BASICs, utilice la función CHR\$ para convertir los números a bytes.

El indicador LED Sync LED de la placa del controlador Mini SSC II puede ayudarle a depurar las rutinas de su puerto serie. Se iluminará de manera continua la primera vez que se alimente la placa y se mantendrá encendido hasta que se reciban la primera instrucción de tres bytes. Por lo tanto, el LED se iluminará sólo después de que se haya recibido un marcador de sincronización y una dirección de servo válidos. Se mantendrá encendido hasta que se reciba un byte de posición y después se apagará. Si su programa envía muchos datos al controlador Mini SSC II, aparentemente el LED se iluminará de manera continua, aunque en realidad estará parpadeando muy rápido.

**Programación de Windows 95/98/NT:** Un DLL de Windows de 32 bits está disponible en nuestra página web en la dirección [www.seetron.com/ssc\\_an1.htm](http://www.seetron.com/ssc_an1.htm). En la documentación se incluye una descripción completa del DLL, más programas de ejemplo para Microsoft Visual BASIC 6 y Borland Delphi 5. Puede utilizar el DLL con cualquier entorno de programación de Windows compatible con DLLs.

#### **Uso de varios controladores Mini SSCs**

Para controlar dos controladores Mini SSC II, conéctelos en paralelo a la misma línea de puerto serie; compruebe las sugerencias acerca del cableado de la siguiente página. Configure ambas unidades con la misma tasa de baudios. Instale un jumper en el zócalo de configuración I de una de las unidades; deje este zócalo abierto en la otra. Las direcciones de los servos son las siguientes:

#### **¿Jumper en I? Números de servos**

no 0-7

sí 8-15



**Sugerencias acerca del cableado de múltiples controladores Mini SSC II:** Una vez que tenga uno de los controladores Mini SSC II conectado a un ordenador, la conexión de las unidades adicionales es sencilla. Simplemente deberá conectar los pines S y G del primer Mini SSC II a los pines S y G del segundo. Haga lo mismo con el segundo y tercero, tercero y cuarto, etc. Dado que hay dos pines S y dos pines G en cada controlador Mini SSC II, las conexiones son bastante sencillas.

Para hacer estas conexiones sin soldaduras, utilice cables de jumpers como los de Scott Edwards Electronics, Inc. Estos jumpers tienen unas clavijas que encajan a la perfección en la placa del controlador Mini SSC II (o cualquier otro 0.025").

**Más de 16 servos:** Con un jumper instalado en I, el controlador Mini SSC II suma 8 a las direcciones de los servos, para que la dirección 0 de los servos sea 8, el servo 1 sea 9...y el servo 7 sea 15. Si necesita controlar más de 16 servos, puede utilizar controladores Mini SSC II especiales con rangos de direcciones más grandes del mismo fabricante. Por ejemplo, su segundo o tercer controlador Mini SSC II tendría una dirección de base (la dirección correspondiente al servo 0 sin un jumper I) de 16. El número de las partes de estos controladores Mini SSC II especiales es SSC-ASD2- $n$ , donde  $n$  es el número base del servo. En la siguiente tabla encontrará los primeros cinco números de partes y los números de rangos cubiertos por los servos.

Números de partes	Número de servos (sin jumper en I)	Número de servos (con jumper en I)
SSC-ASD2 (standard)	0—7	8—15
SSC-ASD2-16	16—23	24—31
SSC-ASD2-32	32—39	40—47
SSC-ASD2-48	48—55	56—63
SSC-ASD2-64	64—71	72—79

### Aspectos teóricos del funcionamiento de los servos

Los servos que funcionan por pulsos proporcionales están diseñados para su uso en coches, barcos y aviones dirigidos por radio control (R/C). Proporcionan un control preciso para realizar los giros, aceleración, rotación, etc...utilizando una señal que es fácil de transmitir y recibir. La señal está constituida por pulsos que oscilan entre 1 y 2 milisegundos de longitud, repetidos 60 veces por segundo. El servo posiciona el eje de salida en proporción al ancho del pulso, tal y como se indica en la Figura 5.

Para las aplicaciones por radio un servo necesita un rango de movimiento más grande de 90°, dado que normalmente las piezas mecánicas no se mueven más allá de los 90°. Por lo tanto cuando envía pulsos dentro del rango de pulsos especificado por el fabricante de 1 a 2 ms, obtendrá un rango de movimiento de 90°. Sin jumper en el zócalo (R)ango, el valor de posición 0 corresponde a un pulso de 1 ms; 254 a 2.016 ms. Un cambio de 1-unidad en el valor de posición produce un cambio de 4- microsegundos en el ancho de los pulsos. La resolución de la posición es 0.36°/unidad (90°/250).

La mayoría de los servos tienen un rango de movimiento mecánico superior a los 90°, aunque es sólo para ajustar las variaciones de los componentes, posición de montaje, etc. El controlador de servos Mini SSC II le permite utilizar este rango extra. Con un jumper en (R)ango, un valor de posición 0 corresponde a un pulso de 0,5-ms; 254 a 2.53 ms. Un cambio de 1-unidad en el valor de posición produce un cambio de 4- microsegundos en el ancho de los pulsos. . La resolución de la posición es 0.72°/unidad (180°/250).

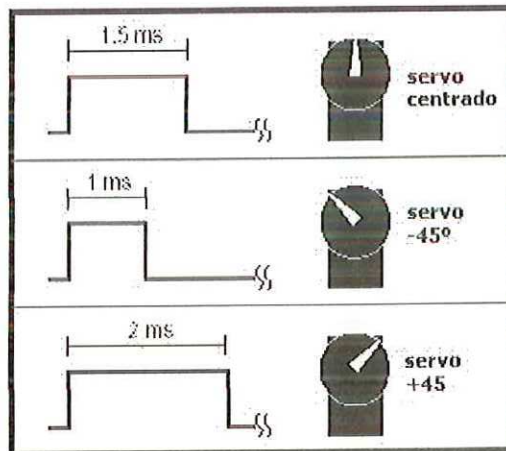


Figure 5. Los servos son controlados por pulsos de 1 a 2 ms repetidos a una frecuencia de 60Hz.

## Ejemplos de programas

```
' Listing 1. BASIC Stamp I Program Demonstrating the Mini SSC
' Program: SCAN.BAS (BS1 servo control demo)
' This program demonstrates servo control using the MiniSSC.
' It commands servo 0 slowly and smoothly through its full
' range of travel. To run this program, leave all configuration
' jumpers off the Mini SSC board. Connect:
' BS1 Mini SSC Purpose
' -----
' pin0 S(in) pin S Serial signal
' Vss S(in) pin G Ground
' Plug a servo into Mini SSC output 0 and connect power as described
' in the manual. Run this program. The servo will slowly scan back
' and forth. Change for/next step values to change servo movement.
SYMBOL svo = 0 ' Use servo 0.
SYMBOL sync = 255 ' Mini SSC sync byte.
SYMBOL pos = b2 ' Byte variable b2 holds position value.
again:
for pos = 0 to 254 step 1 ' Rotate clockwise in 1-unit steps.
serout 0,n2400,(sync,svo,pos) ' Command the Mini SSC.
next pos ' Next position.
for pos = 254 to 0 step -1 ' Rotate counter-clock, 1-unit steps.
serout 0,n2400,(sync,svo,pos) ' Command the mini SSC.
next pos ' Next position.
goto again ' Do it again.
```

```
' Listing 2. BASIC Stamp II Program Demonstrating the Mini SSC
' Program: SCAN.BS2 (BS2 servo control demo)
' This program demonstrates servo control using the MiniSSC.
' It commands servo 0 slowly and smoothly through its full
' range of travel. To run this program, install a jumper at
' B on the Mini SSC board; leave all other jumpers off.
' Connect:
' BS2 Mini SSC Purpose
' -----
' P0 S(in) pin S Serial signal
' Vss S(in) pin G Ground
' Plug a servo into Mini SSC output 0 and connect power as described
' in the manual. Run this program. The servo will slowly scan back
' and forth. Try changing the step values in the for/next loops
' to see the effect on servo movement.
svo con 0 ' Use servo 0.
sync con 255 ' Mini SSC sync byte.
pos var byte ' Byte variable holds position value.
n96n con $4054 ' Baudmode: 9600 baud (BS2-SX, change to $40F0).
n24n con $418D ' Baudmode: 2400-baud (BS2-SX, change to $43FD).
```

```
' BS2 program example, continued.
again:
for pos = 0 to 254 step 1 ' Rotate clockwise in 1-unit steps.
serout 0,n96n,[sync,svo,pos] ' Command the mini SSC.
next ' Next position.
for pos = 254 to 0 step 1 ' Rotate counter-clock, 1-unit steps.
serout 0,n96n,[sync,svo,pos] ' Command the mini SSC.
next ' Next position.
goto again ' Do it again.
```

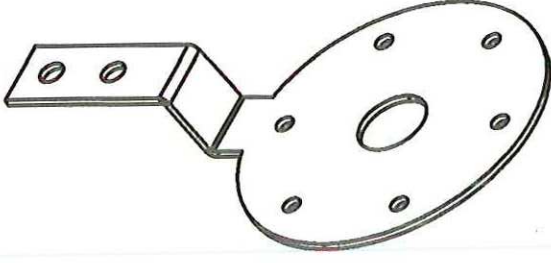
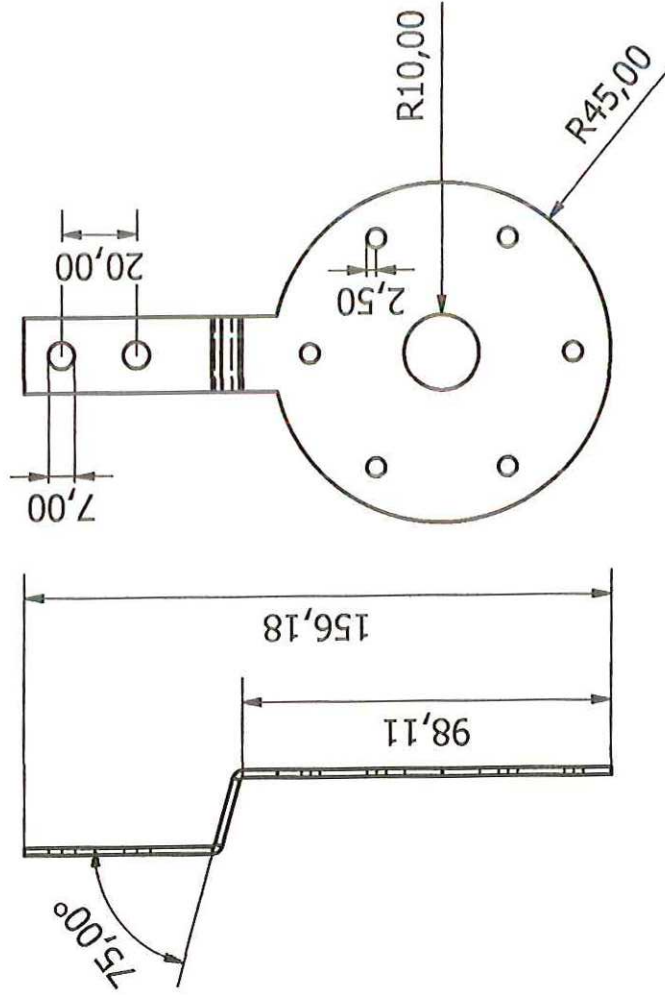
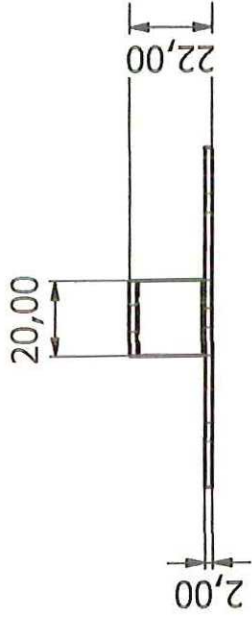
---

```
' Listing 3. QBASIC Program for PCs Demonstrating the Mini SSC
' Note:This program is written for QBASIC, an old, but still common
' MS-DOS programming language. For programming under Windows 95 or
' greater with Visual BASIC see www.seetron.com/ssc\_an1.htm .
DEFINT A-Z
Sync.byte = 255
' The line below assumes that the B jumper is installed for
' 9600-baud operation.
OPEN "com1:9600,N,8,1,CD0,CS0,DS0,OP0" FOR OUTPUT AS #1
CLS
PRINT " MINI SERIAL SERVO CONTROLLER"
PRINT : PRINT
PRINT "At the prompt, type the servo number (0 to 7), a comma,
PRINT "and a position value (0 to 254).\"
PRINT "Press <CNTRL> - <Break> to end.\"
Again:
LOCATE 8, 1
PRINT " \"
LOCATE 8, 1
INPUT "Servo,position>\", Servo, Position
' Perform some basic error trapping
IF Servo > 7 THEN Servo = 7
IF Servo < 0 THEN Servo = 0
IF Position > 254 THEN Position = 254
IF Position < 0 THEN Position = 0
PRINT #1, CHR$(Sync.byte); CHR$(Servo); CHR$(Position);
GOTO Again
```

---

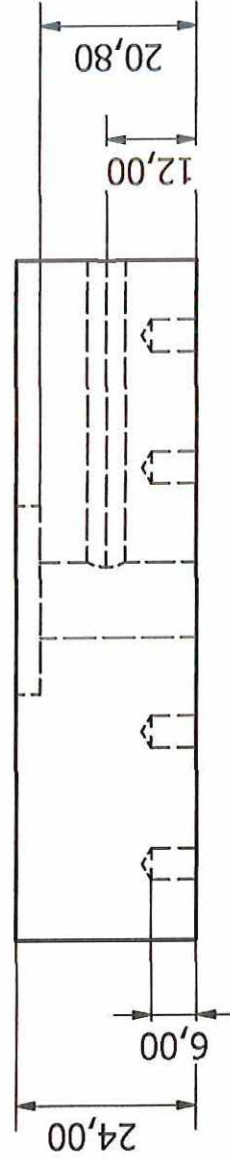
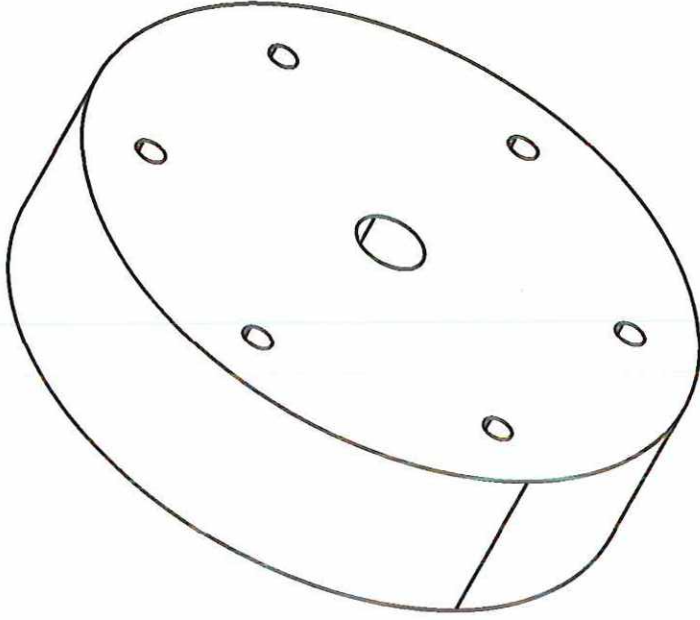
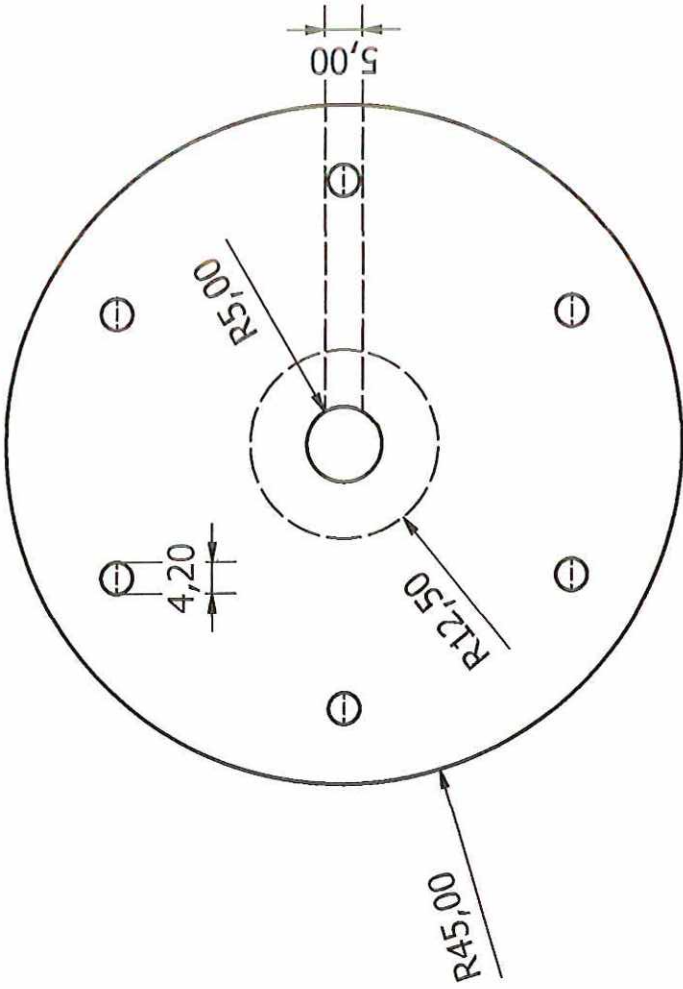
**Si desea obtener más ejemplos de programas para el controlador Mini SSC y consultar proyectos de otros usuarios visite [www.seetron.com/ssc.htm](http://www.seetron.com/ssc.htm). También puede utilizar el software de control de servos gratuito disponible en <http://www.superrobotica.com/VisualSC.htm>**

**ANEXO E. Planos de los diseños mecánicos**



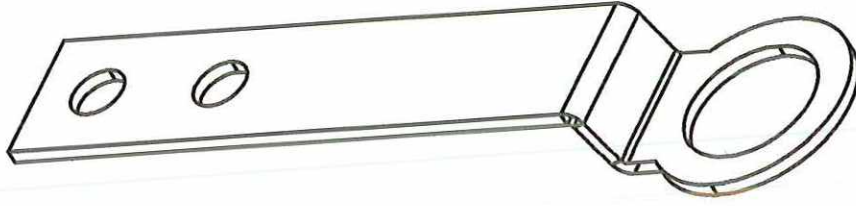
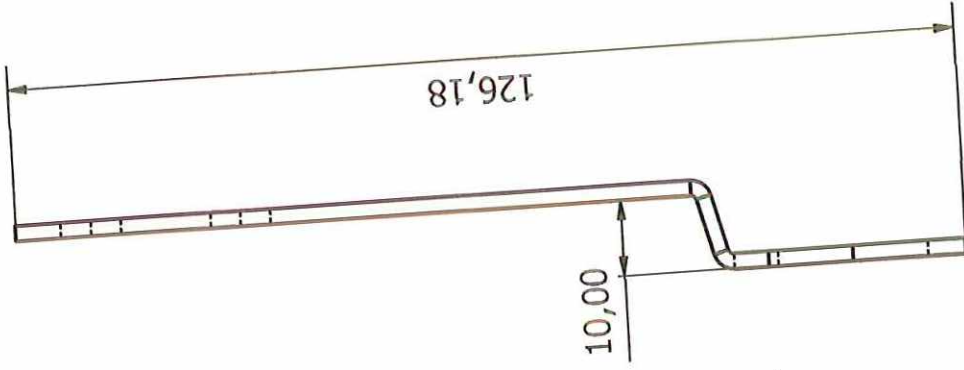
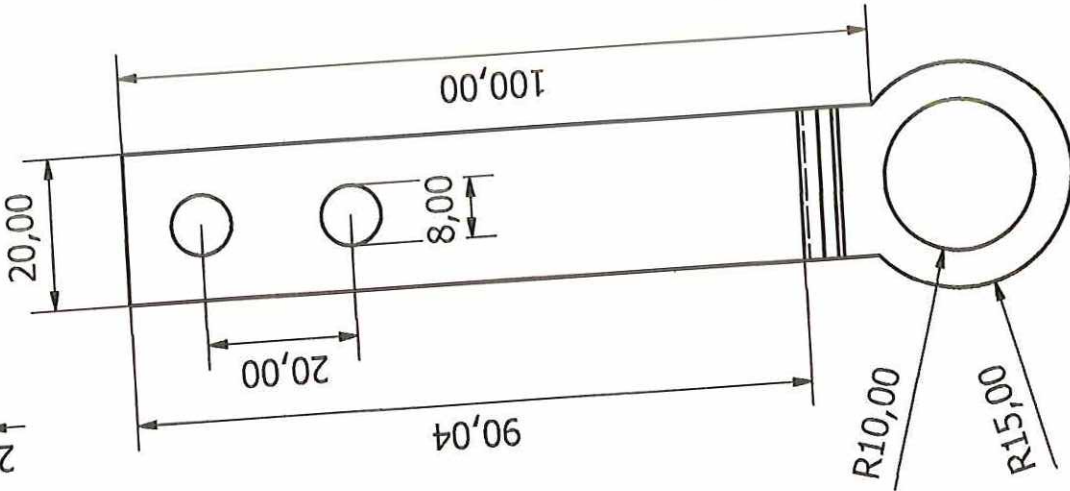
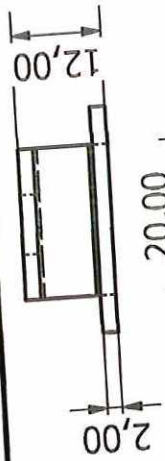
Escala 1:2  
 Unidades en [mm]

Diseño de Juan Grosso	Revisado por	Aprobado por	Fecha	Fecha
			30/05/2008	
PROYECTO DE GRADO DE INGENIERÍA MECATRÓNICA			EXOSQUELETO	
			Edición	Hoja
			1	1 / 1
			Platina para motor	



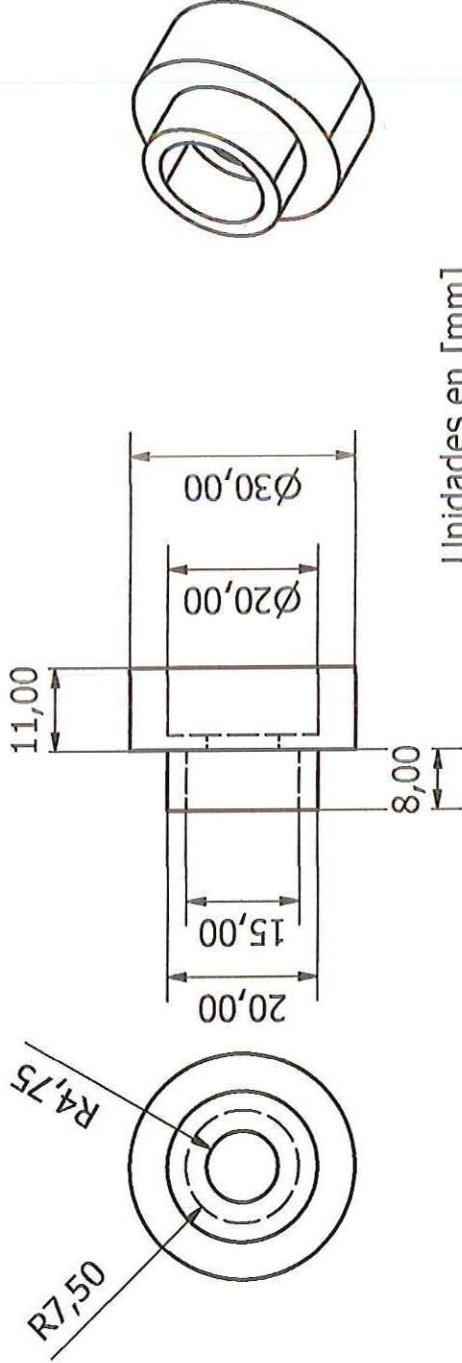
Unidades en [mm]

Diseño de Juan Grosso	Revisado por	Aprobado por	Fecha 30/05/2008
PROYECTO DE GRADO DE INGENIERÍA MECATRÓNICA		EXOSQUELETO	
Base acople motor		Edición 1	Hoja 1 / 1

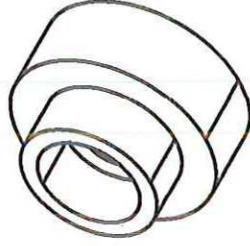


Diseño de Juan Grosso	Revisado por	Aprobado por	Fecha 30/05/2008	Fecha	Edición 1	Hoja 1 / 1
PROYECTO DE GRADO DE INGENIERÍA MECATRÓNICA			EXOSQUELETO			Platina para articulación



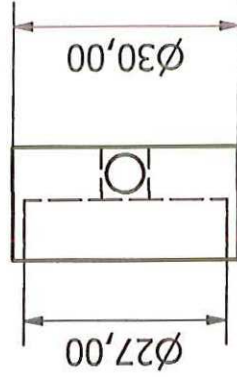
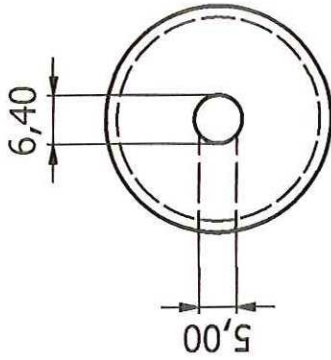
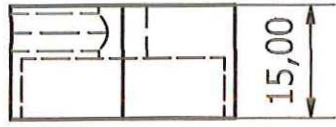


Unidades en [mm]



Diseño de Juan Grosso	Revisado por	Aprobado por	Fecha	Fecha
			30/05/2008	
PROYECTO DE GRADO DE INGENIERÍA MECATRÓNICA			EXOSQUELETO	
			Articulación rotacional - 2	Edición 1
				Hoja 1 / 1

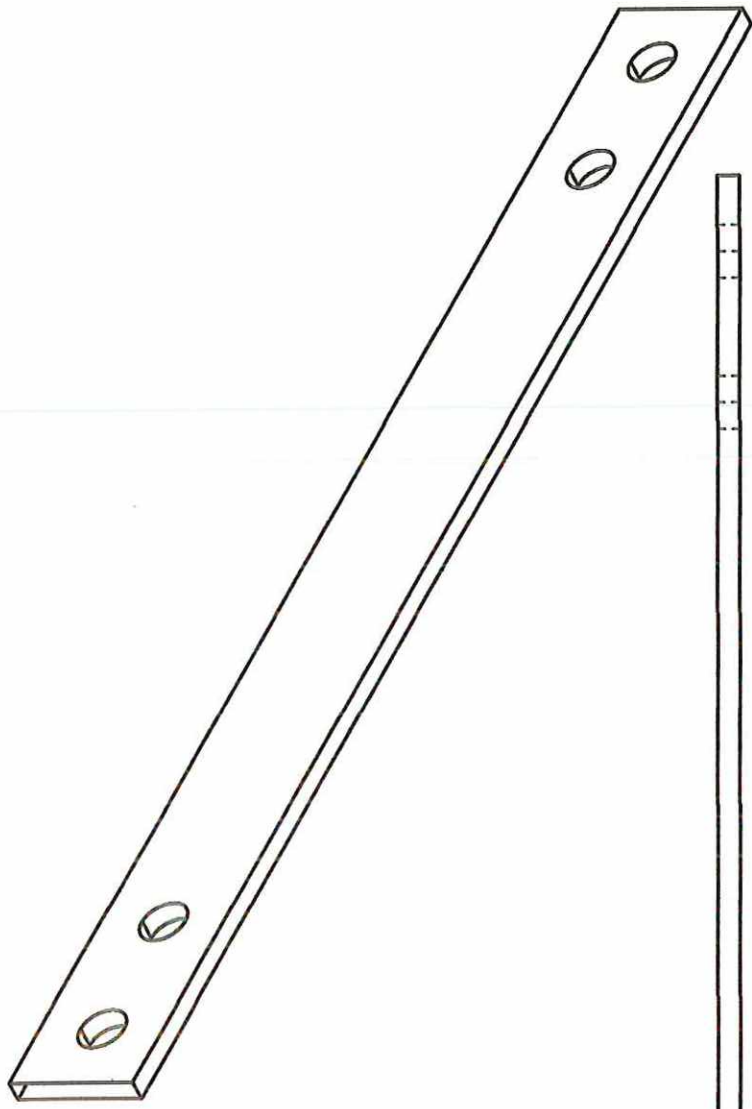
Z W I A



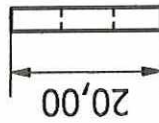
Unidades en [mm]

Diseño de Juan Grosso	Revisado por	Aprobado por	Fecha	Fecha
			30/05/2008	
PROYECTO DE GRADO DE INGENIERÍA MECATRÓNICA			EXOSQUELETO	
			Articulación rotacional-1	Edición
				Hoja
				1 / 1

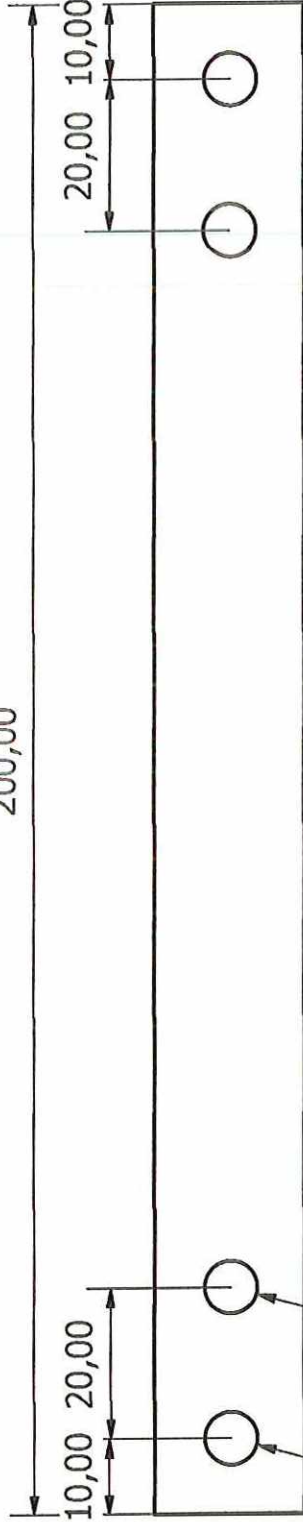
W



200,00



20,00



10,00 20,00

20,00 10,00

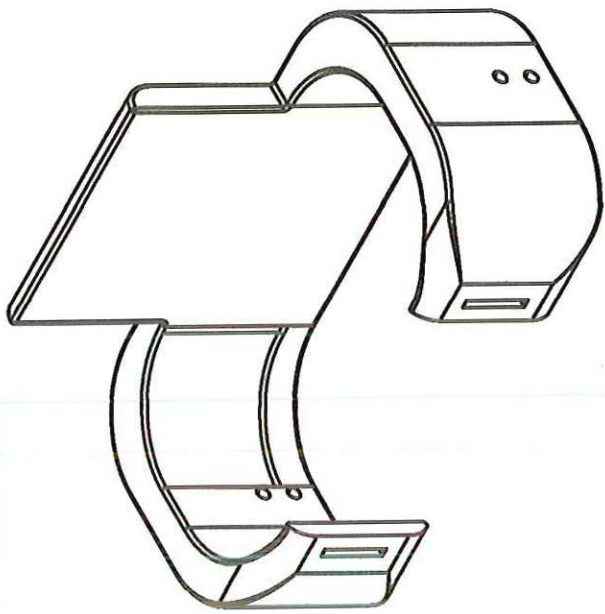
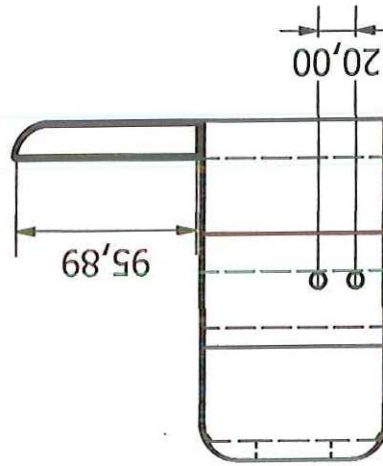
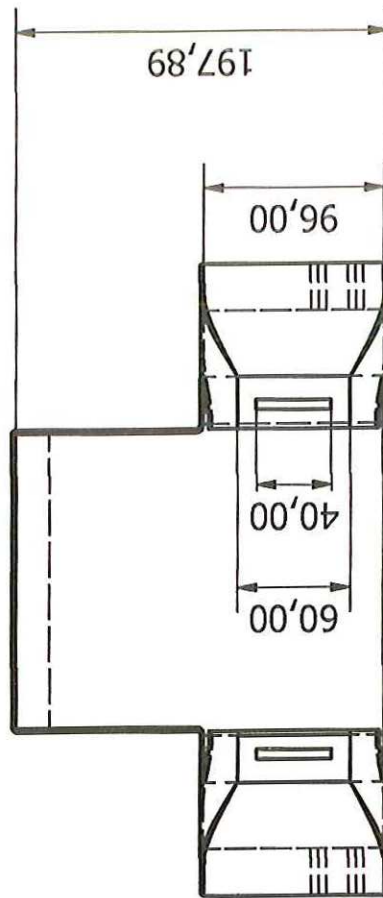
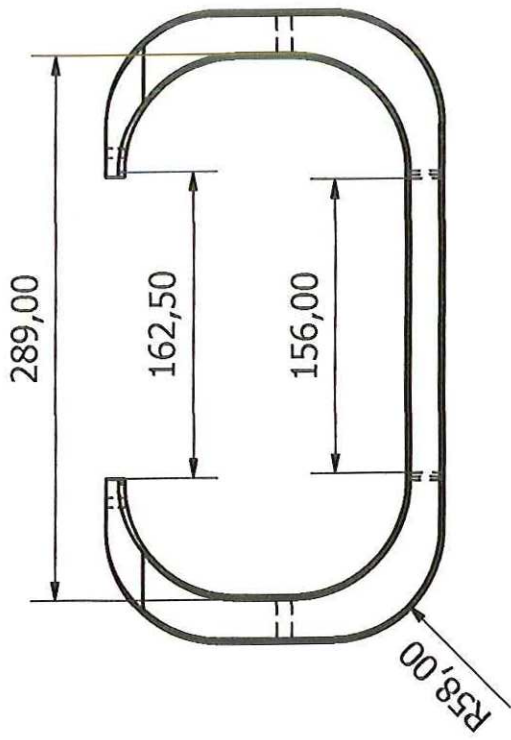
R3,50

R3,50

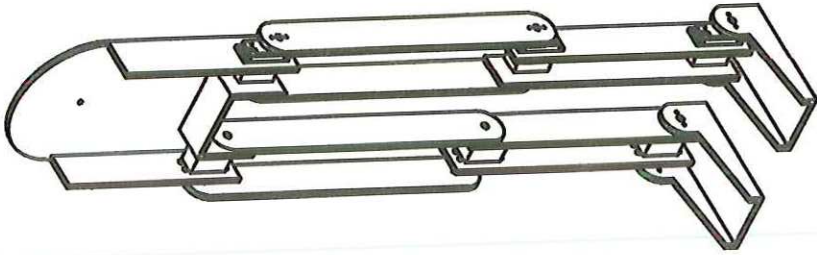
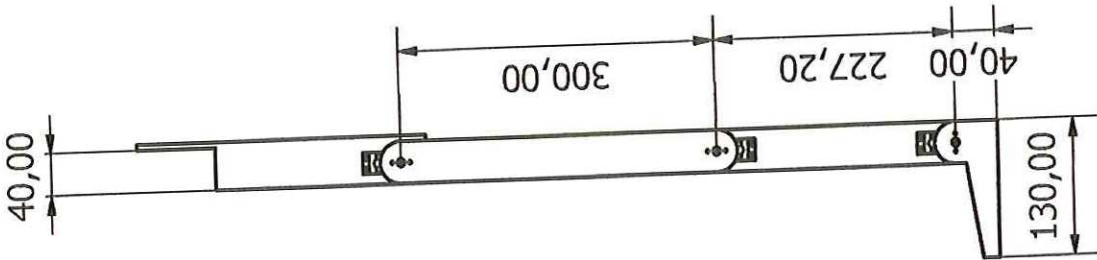
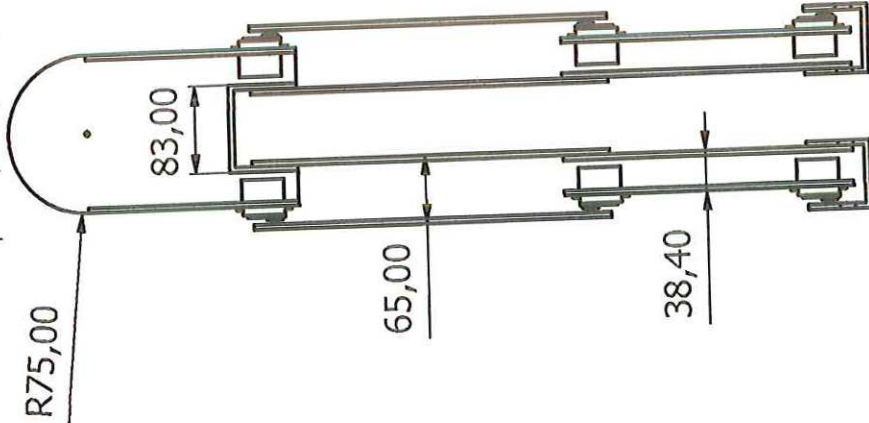
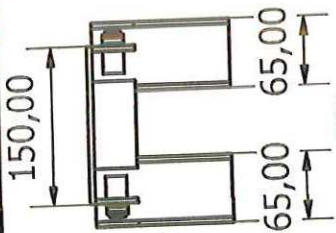
Diseño de Juan Grosso	Revisado por	Aprobado por	Fecha 30/05/2008
PROYECTO DE GRADO DE INGENIERÍA MECATRÓNICA		EXOSQUELETO	
		Lamina de acople	Edición 1
			Hoja 1 / 1

Unidades en [mm]

W

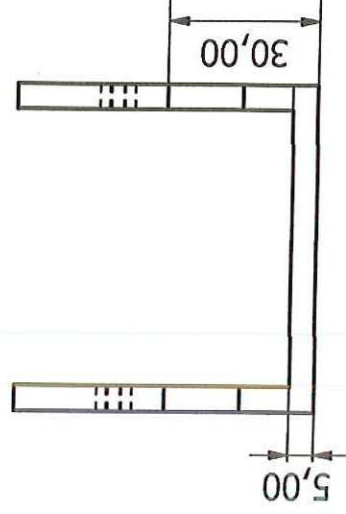
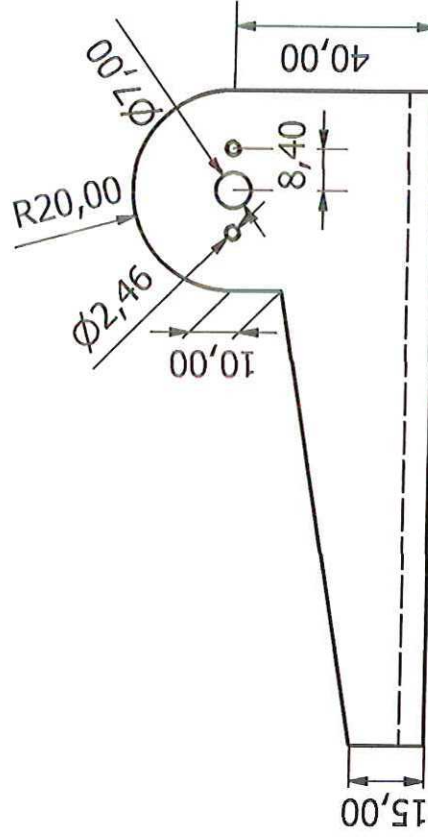
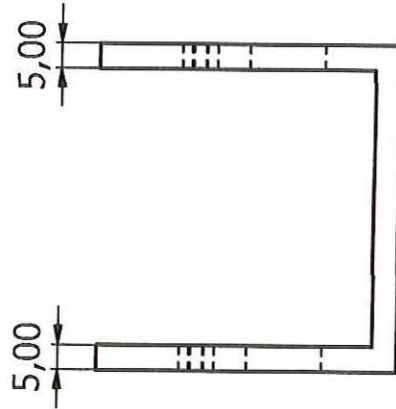
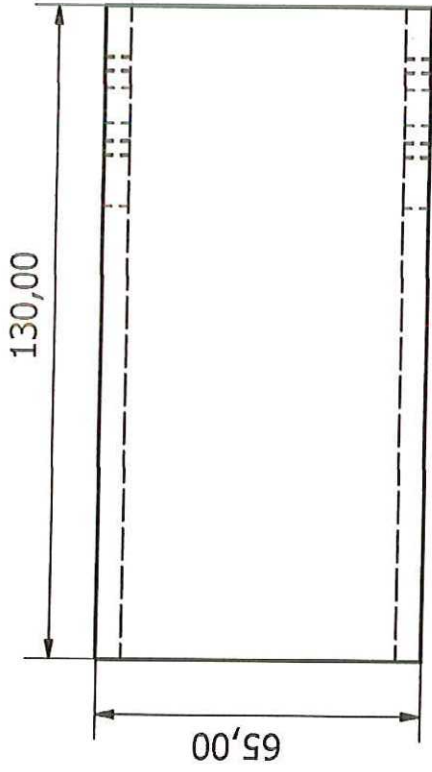
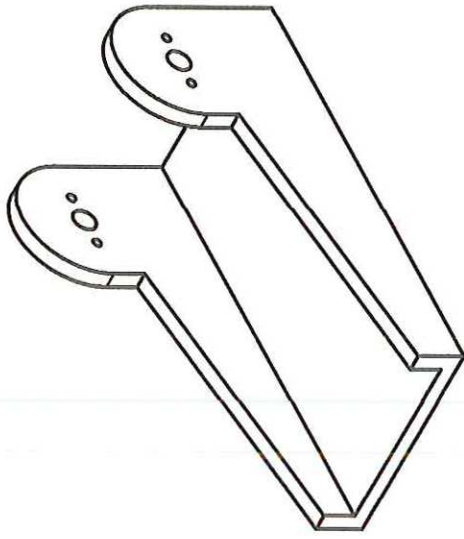


Diseño de Juan Grosso	Revisado por	Aprobado por	Fecha	Fecha
			30/05/2008	
PROYECTO DE GRADO DE INGENIERÍA MECATRÓNICA			EXOSQUELETO	
			Edición	Hoja
			1	1 / 1
			Soporte Cadema	



Diseño de Juan Grosso	Revisado por	Aprobado por	Fecha	Fecha
			13/05/2008	
PROYECTO DE GRADO DE INGENIERIA MECATRÓNICA			PLATAFORMA BIPEDA	
			Edición	Hoja
			1	1 / 1
			Ensamble	

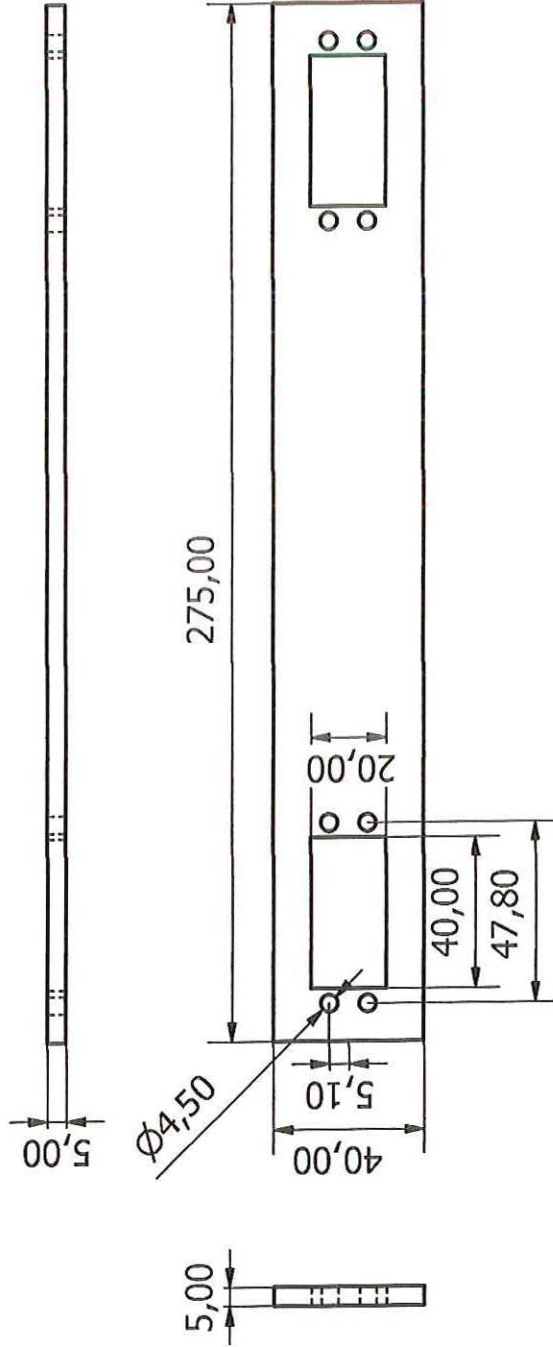
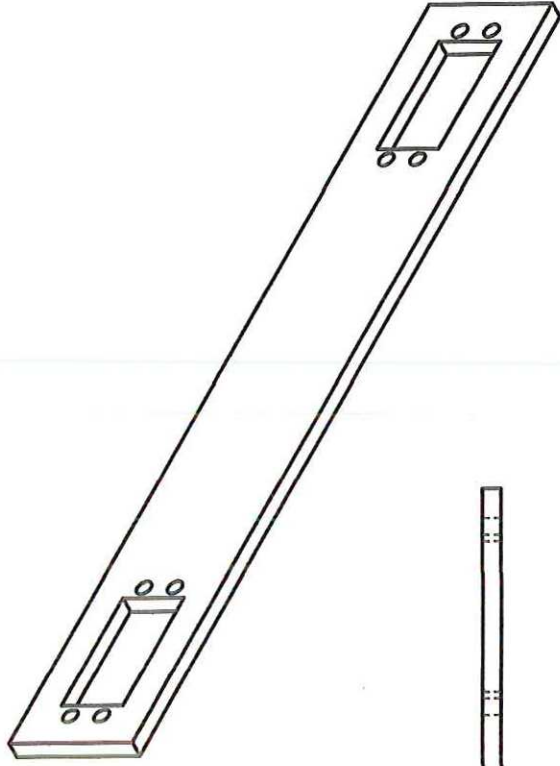
W



Unidades en [mm]

Diseño de Juan Grosso	Revisado por	Aprobado por	Fecha	Fecha
			13/05/2008	
PROYECTO DE GRADO INGENIERIA MECATRONICA			PIE DE LA PLATAFORMA BÍPEDA	
Pie-1			Edición	Hoja
			1	1 / 1

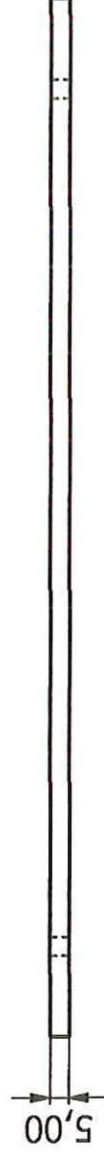
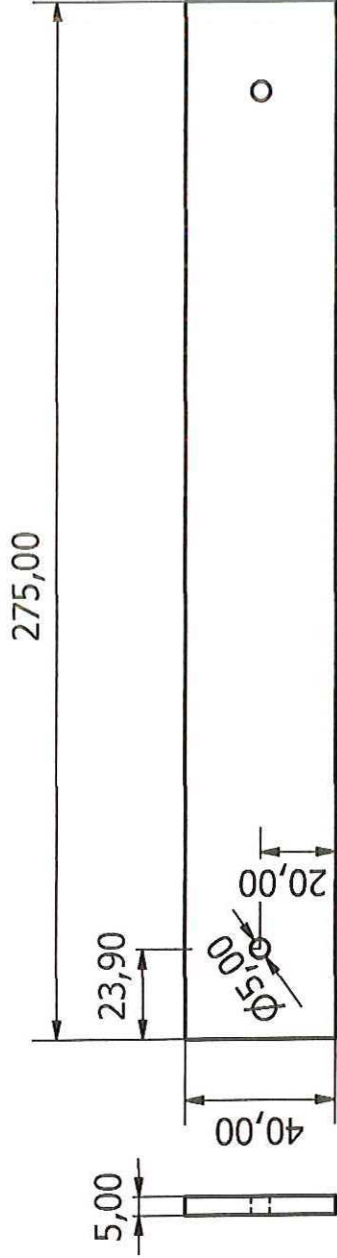
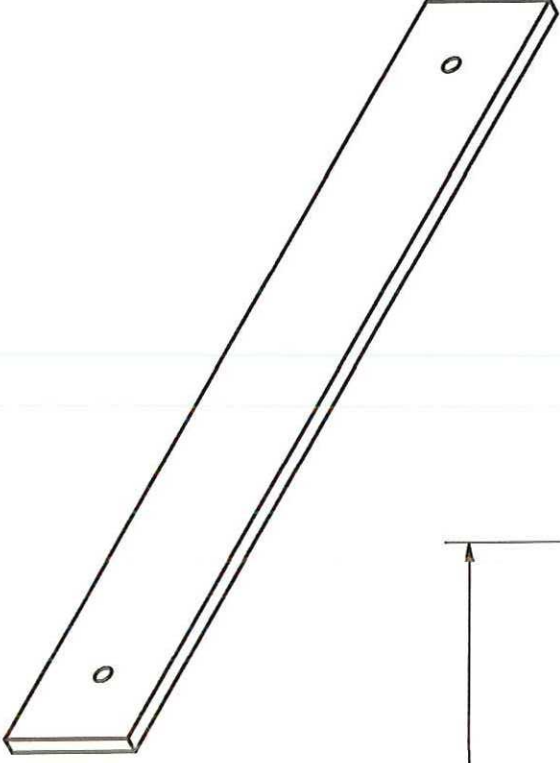
W



Unidades en [mm]

Diseño de Juan Grosso	Revisado por	Aprobado por	Fecha 13/05/2008
PROYECTO DE GRADO DE INGENIERÍA MECATRÓNICA		PANTORRILLA PLATAFORMA BIPEDA	
Pantorrilla-1		Edición	Hoja 1 / 1

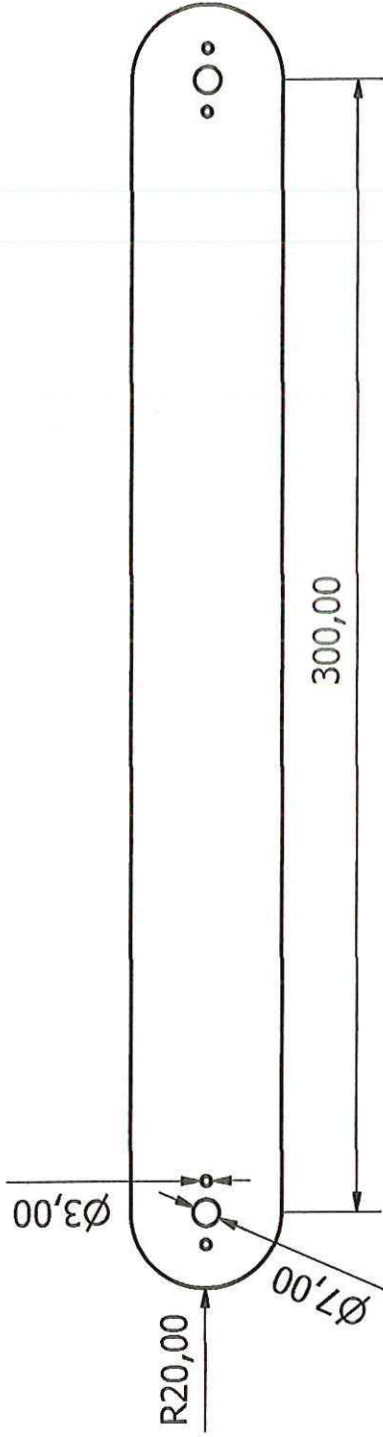
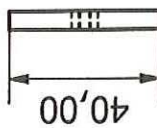
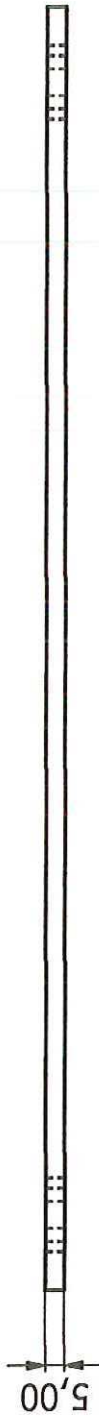
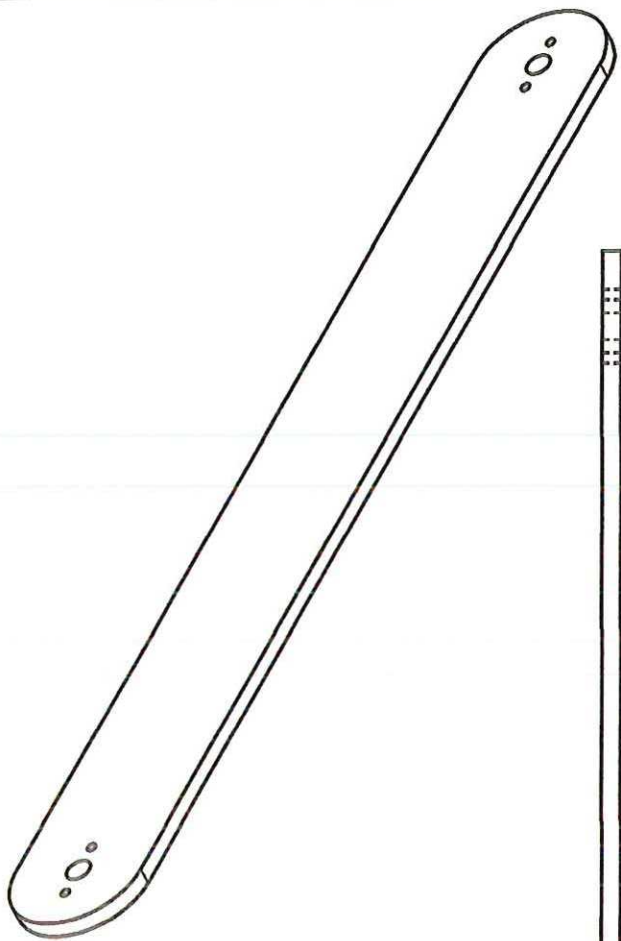




Unidades en [mm]

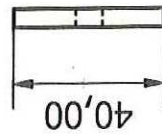
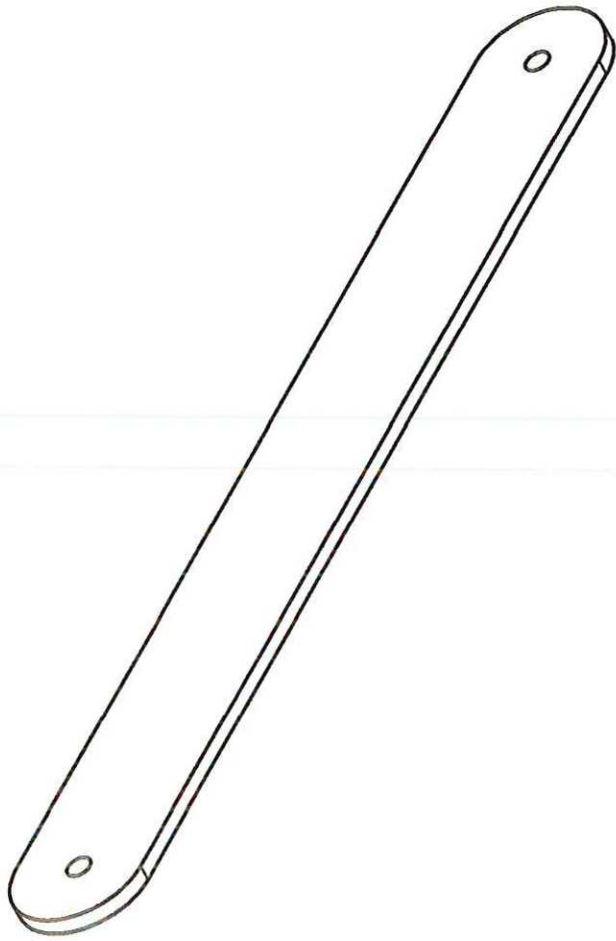
Diseño de Juan Grosso	Revisado por	Aprobado por	Fecha	Fecha 13/05/2008
PROYECTO DE GRADO DE INGENIERÍA MECATRÓNICA			PANTORRILLA DE PLATAFORMA BIPEDA	
Pantorrilla-2			Edición 1	Hoja 1 / 1





Unidades en [mm]

Diseño de Juan Grosso	Revisado por	Aprobado por	Fecha 13/05/2008
PROYECTO DE GRADO DE INGENIERÍA MECATRÓNICA		Muslo-1	MUSLO-PLATAFORMA BIPEDA
		Edición 1	Hoja 1 / 1



$\varnothing 7,00$

R20,00

300,00

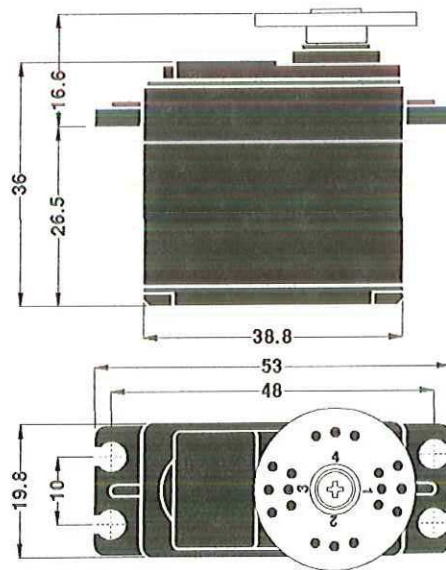
Unidades en [mm]

Diseño de Juan Grosso	Revisado por	Aprobado por	Fecha	Fecha
			13/05/2008	
PROYECTO DE GRADO DE INGENIERÍA MECATRÓNICA			MUSLO-PLATAFORMA BIPEDA	
Muslo-2			Edición 1	Hoja 1 / 1

**ANEXO F. Especificaciones de los servomotores**

## ANNOUNCED SPECIFICATION OF HS-475HB STANDARD DELUXE SERVO

<b>1. TECHNICAL VALUE</b>		
CONTROL SYSTEM	: +PULSE WIDTH CONTROL 1500usec NEUTRAL	
OPERATING VOLTAGE RANGE	: 4.8V TO 6.0V	
TEST VOLTAGE	: AT 4.8V	AT 6.0V
OPERATING SPEED	: 0.23sec/60? AT NO LOAD	0.18sec/60? AT NO LOAD
STALL TORQUE	: 4.4kg.Cm(61.10oz.In)	5.5kg.Cm(76.37oz.In)
IDLE CURRENT	: 7.4mA AT STOPPED	7.7mA AT NO LOAD
RUNNING CURRENT	: 160mA/60? AT NO LOAD	180mA/60? AT NO LOAD
STALL CURRENT	: 900mA	1100mA
DEAD BAND WIDTH	: 5usec	5usec
OPERATING TRAVEL	: 40? /ONE SIDE PULSE TRAVELING 400usec	
DIRECTION	: CLOCK WISE/PULSE TRAVELING 1500 TO 1900usec	
MOTOR TYPE	: CORED METAL BRUSH	
POTENTIOMETER TYPE	: 6 SLIDER/INDIRECT DRIVE	
AMPLIFIER TYPE	: ANALOG CONTROLLER & TRANSISTOR DRIVER	
DIMENSIONS	: 38.8x19.8x36mm(1.52x0.77x1.41in)	
WEIGHT	: 40g(1.41oz)	
BALL BEARING	: TOP/MR106	
GEAR MATERIAL	: HEAVY DUTY RESIN	
HORN GEAR SPLINE	: 24 SEGMENTS/?5.76	
SPLINED HORNS	: REGULAR/R-C, R-D, R-I, R-O, R-X, SUPER/R-XA	
CONNECTOR WIRE LENGTH	: 300mm(11.8in)	
CONNECTOR WIRE STRAND COUNTER	: 60EA	
CONNECTOR WIRE GAUGE	: 22AWG	



### 2. FEATURES

HEAVY DUTY RESIN GEARS, TOP BALL BEARING

### 3. APPLICATIONS

AIRCRAFT 20-60 SIZE, 30 SIZE HELICOPTER, STEERING AND THROTTLE SERVO, TRUCK AND BOATS

### 4. ACCESSORY & OPTION

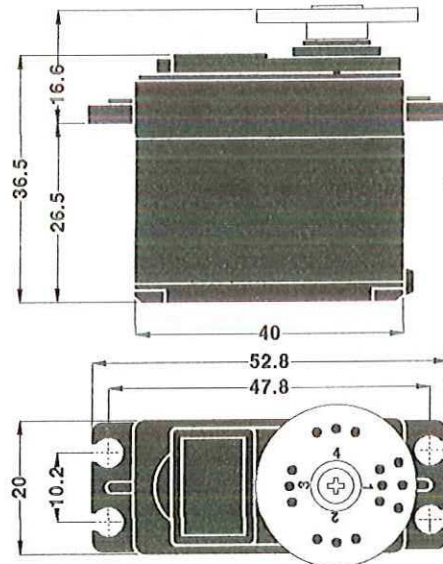
<b>CASE SET/</b>	<b>GEAR SET/</b>	<b>BALL BEARING SET/</b>	<b>HORN SET/</b>
HS475T:1EA	HS475G1:1EA	MR106:1EA	R-C:1EA
HS475M:1EA	HS475G2:1EA	CU 7.99x6:1EA	R-D:1EA
HS475L:1EA	HS475G3:1EA	FE 10x8:1EA	R-I:1EA
PH/T-2 2x30 NI:4EA	HS475G4:1EA		R-O:1EA
	HS75PG:1EA		R-X:1EA
			R-XA:1EA
			WH/W 2.1x15 NI:4EA
			BST 3x5.5:4EA
			NBR 9x6.5x6:4EA

**HITEC RCD KOREA INC.**

# ANNOUNCED SPECIFICATION OF HS-311 STANDARD SERVO

## 1. TECHNICAL VALUE

CONTROL SYSTEM	:+PULSE WIDTH CONTROL 1500usec NEUTRAL	
OPERATING VOLTAGE RANGE	:4.8V TO 6.0V	
TEST VOLTAGE	:AT 4.8V	AT 6.0V
OPERATING SPEED	:0.19sec/60□ AT NO LOAD	0.15sec/60□ AT NO LOAD
STALL TORQUE	:3.0kg.cm(42oz.in)	3.5kg.cm(48.60oz.in)
IDLE CURRENT	:7.4mA AT STOPPED	7.7mA AT STOPPED
RUNNING CURRENT	:160mA/60□ AT NO LOAD	180mA/60□ AT NO LOAD
STALL CURRENT	:700mA	800mA
DEAD BAND WIDTH	:5usec	5usec
OPERATING TRAVEL	:40□/ONE SIDE PULSE TRAVELING 400usec	
DIRECTION	:CLOCK WISE/PULSE TRAVELING 1500 TO 1900usec	
MOTOR TYPE	:CORED METAL BRUSH	
POTENTIOMETER TYPE	:4 SLIDER/DIRECT DRIVE	
AMPLIFIER TYPE	:ANALOG CONTROLLER & TRANSISTOR DRIVER	
DIMENSIONS	:40x20x36.5mm(1.57x0.78x1.43in)	
WEIGHT	:43g(1.51oz)	
BALL BEARING	:TOP/RESIN BUSHING	
GEAR MATERIAL	:RESIN	
HORN GEAR SPLINE	:24 SEGMENTS/□5.76	
SPLINED HORNS	:SUPER/R-XA	
CONNECTOR WIRE LENGTH	:300mm(11.81in)	
CONNECTOR WIRE STRAND COUNTER	:40EA	
CONNECTOR WIRE GAUGE		



## 2. FEATURES

LONG LIFE POTENTIOMETER, TOP RESIN BUSHING

## 3. APPLICATIONS

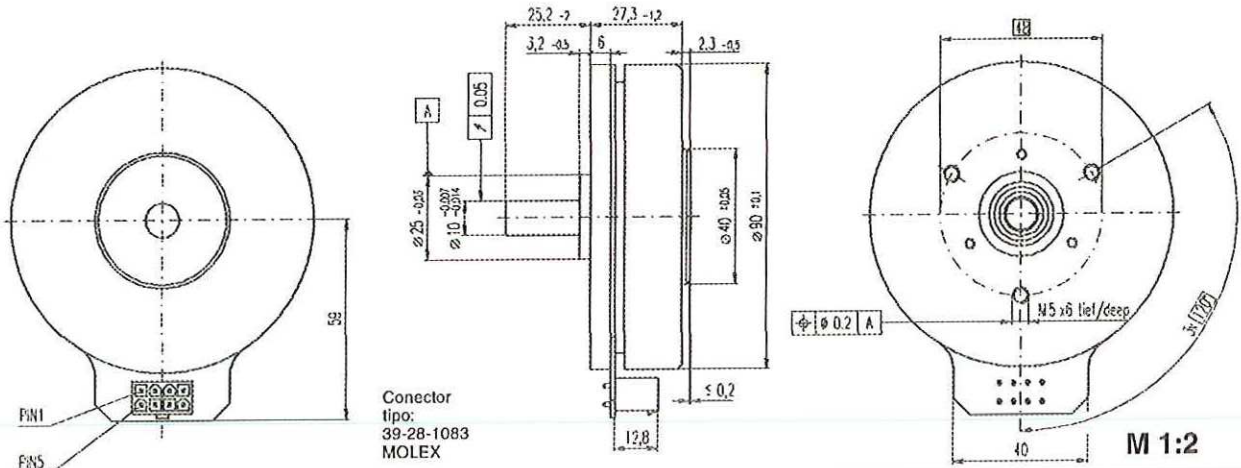
AIRCRAFT 20-40 SIZE, STEERING AND THROTTLE SERVO FOR CARS, TRUCK AND BOATS

## 4. ACCESSORY & OPTION

CASE SET/	GEAR SET/	HORN SET/
HS322T:1EA	HS322G1:1EA	R-XA:1EA
HS322M:1EA	HS322G2:1EA	
HS322L:1EA	HS322G3:1EA	
PH/T-2 2x30 NI:4EA	HS322G4:1EA	
	HS300RB:1EA	

**HITEC RCD KOREA INC.**

# EC 90 flat Ø90 mm, Conmutación electrónica (Brushless), 90 Vatios



Conector  
tipo:  
39-28-1083  
MOLEX

maxon flat motor

- Programa Stock
- Programa Estándar
- Programa Especial (previo encargo)

## Números de Referencia

		con sensores Hall	
		10.35.2	20.35.2
<b>Datos del motor (provisionales)</b>			
Valores a tensión nominal			
1	Tensión nominal	V	24.0
2	Velocidad en vacío	rpm	3190
3	Corriente en vacío	mA	539
4	Velocidad nominal	rpm	2650
5	Par nominal (máx. par permanente)	mNm	387
6	Corriente nominal (máx. corriente en continuo)	A	5.39
7	Par de arranque	mNm	4670
8	Corriente de arranque	A	66.2
9	Máx. rendimiento	%	83
<b>Características</b>			
10	Resistencia en bornes fase-fase	Ω	0.363
11	Inductancia en bornes fase-fase	mH	0.264
12	Constante de par	mNm / A	70.5
13	Constante de velocidad	rpm / V	135
14	Relación velocidad / par	rpm / mNm	0.697
15	Constante de tiempo mecánica	ms	22.3
16	Inercia del rotor	gcm <sup>2</sup>	3060

## Especificaciones

- Datos térmicos**
- 17 Resistencia térmica carcasa/ambiente 1.89 K/W
  - 18 Resistencia térmica bobinado/carcasa 2.99 K/W
  - 19 Constante de tiempo térmica del bobinado 52.6 s
  - 20 Constante de tiempo térmica del motor 281 s
  - 21 Rango de funcionamiento -40 ... +100°C
  - 22 Máx. temperatura del bobinado +125°C
- Datos mecánicos (rodamiento a bolas pretensado)**
- 23 Máx. velocidad permitida 5000 rpm
  - 24 Juego axial < 15 N 0 mm
  - con carga axial > 15 N 0.14 mm
  - 25 Juego radial pretensado
  - 26 Carga axial máx. (dinámica) 12 N
  - 27 Máx. fuerza de empuje a presión (estática) 150 N
  - (idem, con eje sostenido) 8000 N
  - 28 Carga radial máx. a 7.5 mm de la brida 30 N
- Otras especificaciones**
- 29 Número de pares de polos 12
  - 30 Número de fases 3
  - 31 Peso del motor 648 g

Los datos de la tabla son valores nominales.

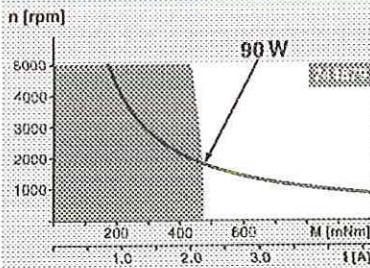
### Conexiones

- Pin 1 Sensor Hall 1
  - Pin 2 Sensor Hall 2
  - Pin 3 4.5 ... 24 VDC
  - Pin 4 Bobinado 3 motor
  - Pin 5 Sensor Hall 3
  - Pin 6 GND
  - Pin 7 Bobinado 1 motor
  - Pin 8 Bobinado 2 motor
- Esquema de conexión para los sensores Hall v. p. 26

### Cables

Cable de la conexión, longitud 500 mm 339380

## Rango de funcionamiento



## Leyenda

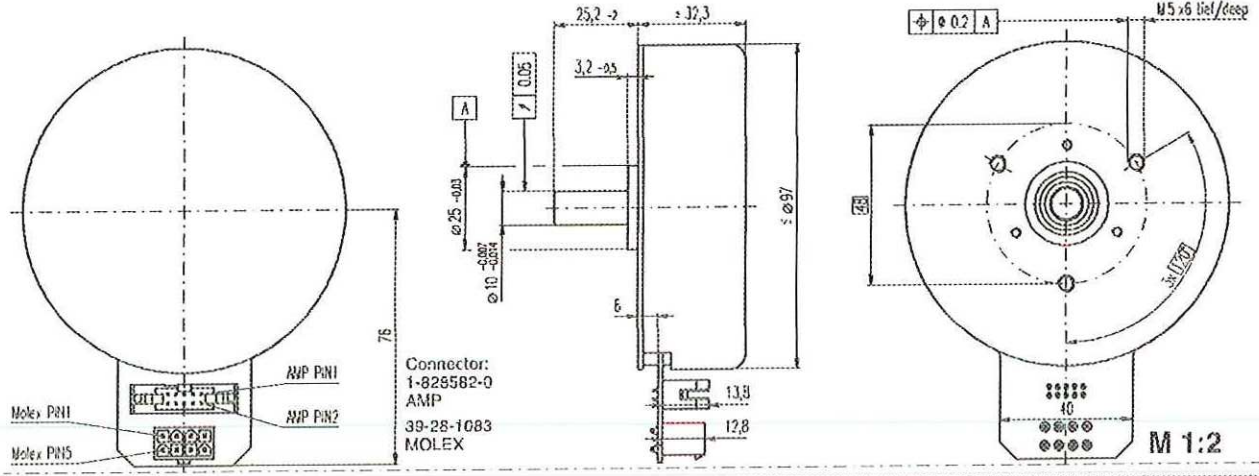
- Funcionamiento continuo**  
Teniendo en cuenta los valores de resistencia térmica antes mencionados (líneas 17 y 18). El rotor alcanzará la máxima temperatura durante funcionamiento continuo a 25°C de temperatura ambiente = límite térmico.
- Funcionamiento intermitente**  
El motor puede ser sobrecargado durante cortos períodos (cíclicamente).
- Potencia nominal asignada**

### Electrónicas Recomendadas:

DEC 50/5	Página 277
DECV 50/5	278
DEC 70/10	278
EPOS 24/5	286
EPOS P 24/5	287
EPOS 70/10	287
Notas	20

# EC 90 flat brushless, 60 Watt, with encoder

**NEW**

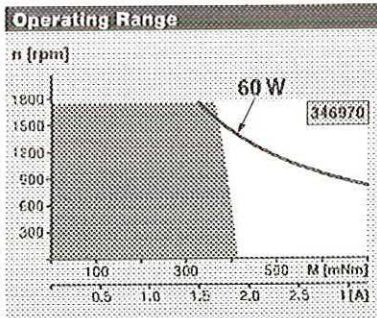


maxon flat motor

<input type="checkbox"/> Stock program <input type="checkbox"/> Standard program <input type="checkbox"/> Special program (on request)	Order Number
--	--------------

Motor Data (provisional)	with hall sensors	353984	346970
<b>Values at nominal voltage</b>			
1 Nominal voltage	V	12.0	36.0
2 No load speed	rpm	1500	1560
3 No load current	mA	338	108
4 Nominal speed	rpm	1260	1240
5 Nominal torque (max. continuous torque)	mNm	327	405
6 Nominal current (max. continuous current)	A	4.56	1.78
7 Stall torque	mNm	2330	3400
8 Starting current	A	33.1	15.5
9 Max. efficiency	%	81.2	84.3
<b>Characteristics</b>			
10 Terminal resistance phase to phase	$\Omega$	0.363	2.3
11 Terminal inductance phase to phase	mH	0.264	2.5
12 Torque constant	mNm/A	70.5	217
13 Speed constant	rpm/V	135	44.0
14 Speed / torque gradient	rpm/mNm	0.697	0.467
15 Mechanical time constant	ms	22.3	15.0
16 Rotor inertia	gcm <sup>2</sup>	3060	3060

Specifications		
<b>Thermal data</b>		
17 Thermal resistance housing-ambient	3.7 K/W	
18 Thermal resistance winding-housing	3.4 K/W	
19 Thermal time constant winding	60 s	
20 Thermal time constant motor	549 s	
21 Ambient temperature	-20 ... +70°C	
22 Max. permissible winding temperature	+125°C	
<b>Mechanical data (preloaded ball bearings)</b>		
23 Max. permissible speed	1750 rpm	
24 Axial play at axial load < 15 N	0 mm	
	> 15 N	0.14 mm
25 Radial play	preloaded	
26 Max. axial load (dynamic)	12 N	
27 Max. force for press fits (static)	150 N	
	(static, shaft supported)	8000 N
28 Max. radial loading, 7.5 mm from flange	30 N	
<b>Other specifications</b>		
29 Number of pole pairs	12	
30 Number of phases	3	
31 Weight of motor	713 g	



**Comments**

- Continuous operation**  
In observation of above listed thermal resistance (lines 17 and 18) the maximum permissible winding temperature will be reached during continuous operation at 25°C ambient.  
= Thermal limit.
- Short term operation**  
The motor may be briefly overloaded (recurring).
- Assigned power rating**

Values listed in the table are nominal.	
<b>Technical Data Encoder</b>	
Function principle	optical
Counts per turn	2048
Number of channels	2
Max. operating frequency	60 kHz
Supply voltage V <sub>DD</sub>	5V ± 10%
Output signal	EIA Standard RS422
Driver used	DS26C31T
Output current per channel	-20 ... +20 mA
Signal rise and fall times (typical, at C <sub>L</sub> = 25pF; R <sub>L</sub> = 1k $\Omega$ )	30 ns
<b>Cable</b>	
Connection cable to EPOS, L = 500 mm	354045
Connection cable Encoder, L = 500 mm	354046

<b>Connection Motor</b>	
Pin 1	Hall sensor 1
Pin 2	Hall sensor 2
Pin 3	4.5 ... 24 VDC
Pin 4	Motor winding 3
Pin 5	Hall sensor 3
Pin 6	GND
Pin 7	Motor winding 1
Pin 8	Motor winding 2
Wiring diagram for Hall sensors see page 29	
<b>Connection Encoder</b>	
Pin 1	N.C.
Pin 2	V <sub>+</sub>
Pin 3	GND
Pin 4	N.C.
Pin 5	A1
Pin 6	A
Pin 7	B1
Pin 8	B
Pin 9	N.C.
Pin 10	N.C.

<b>Recommended Electronics:</b>	
EPOS 24/5	Page 294
EPOS2 50/5	295
EPOS 70/10	295
EPOS P 24/5	297
Notes	20

---

---

**ANEXO G. Códigos fuente de los programas**



## 1. Código fuente de la simulación de la marcha humana en SimMechanics

Diagrama de bloques general:

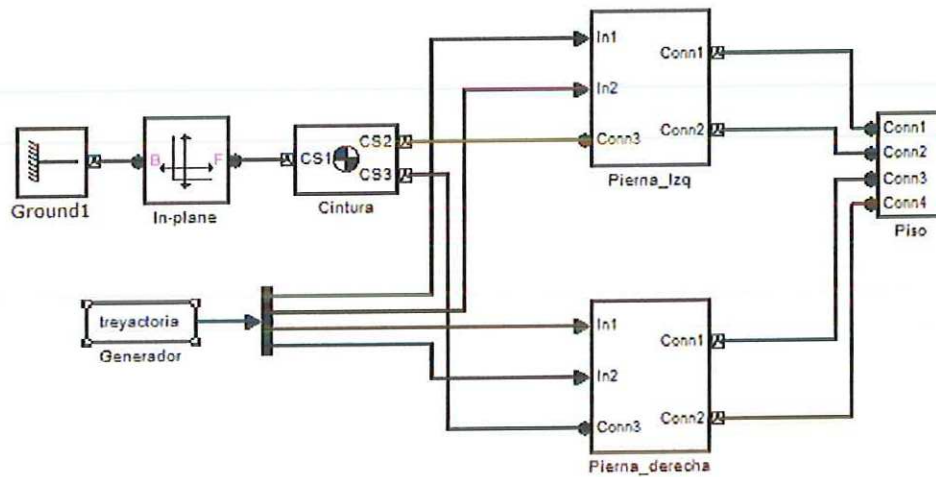


Diagrama de bloques del subsistema pierna izquierda:

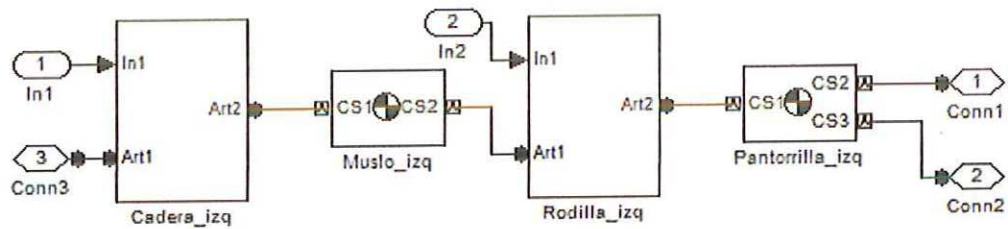
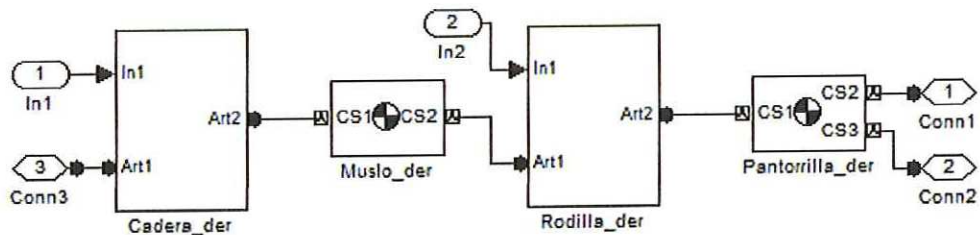


Diagrama de bloques del subsistema pierna derecha:



Los diagramas de bloque de los subsistemas Cadera\_izq y Cadera\_der son exactamente iguales, la diferencia radica en la señal que entra en cada subsistema pues indica la trayectoria deseada para aplicar en dicha articulación. Lo mismo ocurre con los bloques de los subsistemas Rodilla\_izq y Rodilla\_der. De esta manera los subsistemas Cadera y Rodilla se programaron de la siguiente manera:

Diagrama de bloques del subsistema Cadera:

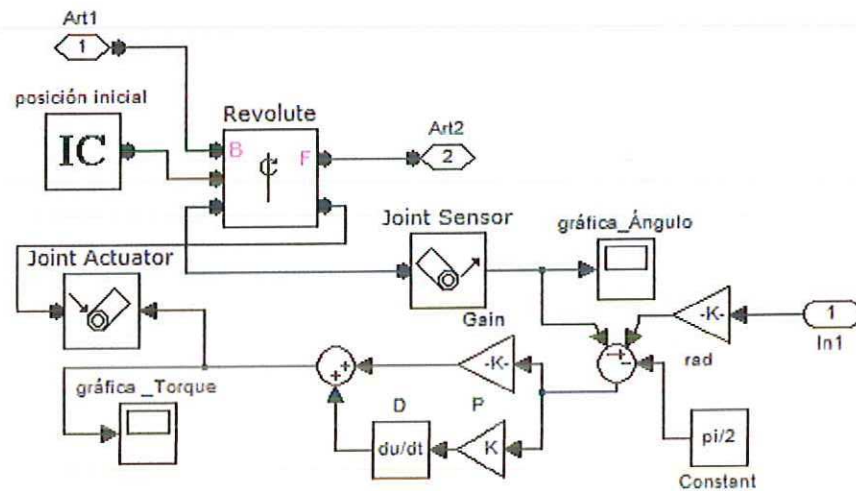
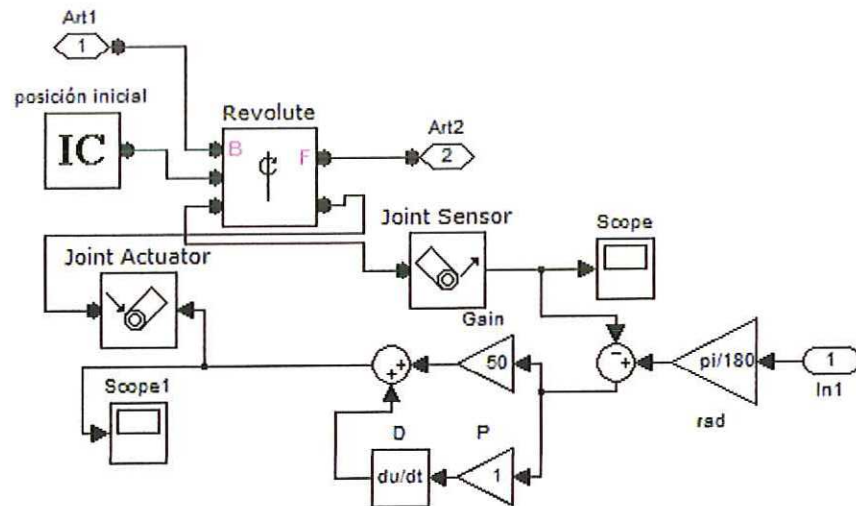


Diagrama de bloques del subsistema Rodilla:



## 2. Código fuente del programa desarrollado para el Microcontrolador

```
*****
; _CONFIG_BODEN_ON & _CP_OFF & _PWRT_OFF & _NDT_OFF & _XT_OSC & _DEBUG_OFF & _CPD_OFF & _LVP_OFF
;*****

list P = 16F873
#include <P16F873.inc>

TEMPH EQU 0x20
TEMP2 EQU 0x21
CONT EQU 0x22
AD_P1 EQU 0x23
AD_P2 EQU 0x24
AD_P3 EQU 0x25
ADCH EQU 0x26

;*****

ORG 0x00
goto MAIN;

ORG 0x04
goto INTER

;*****
; Principal
;*****
MAIN
call CONFIG_ADC
call CONFIG_SERIAL
LAZO
goto LAZO
;*****
; Configuración de módulos ADC y USART
;*****
CONFIG_ADC
call bank1 ; Banco 1
movlw 0x00
movwf ADCON1 ; Justificado a la izquierda
movlw 0xFF
movwf TRISA ; Puerto A como entradas analógicas
call bank0 ; Banco 0
movlw 0x80
movwf ADCON0 ; Fosc/32
clrf AD_P1
clrf AD_P2
clrf AD_P3
return

CONFIG_SERIAL
call bank1 ; Banco 1
bcf TRISC,6 ; RC6/IX salida, pin de transmisión
bsf TRISC,7 ; RC7/Rx entrada, pin de recepción
bsf PIE1,RCIE ; Activamos la máscara particular de interrupciones por recepción
movlw d'25' ; Cargamos el registro SPBRG con d'25' para
movwf SPBRG ; conseguir 2400 baudios con un oscilador de 4MHz; (4MHz/(9600*16))-1 = 25,041
movlw b'10100000' ; Configuramos USART en modo asíncrono y velocidad baja
movwf TXSTA ; todo ello en el registro TXSTA
call bank0 ; Banco 0
movlw b'10010000' ; Habilitamos la recepción continua y tamaño 8 bits
movwf RCSTA ; en el registro RCSTA
movlw b'11000000' ; Habilitamos interrupciones globales
movwf INICON ; y de periféricos
clrf TEMPH
return
```

```

;*****
; Programa para el tratamiento de la interrupción por recepción serial
; Evalúa que el caracter recibido sea "s" e inicia el barrido de asignación de direcciones
; para seleccionar el canal de interés y digitalizarlo a 8 bits.
;*****
INTER
    movlw    b'00000110'    ; Chequea posibles errores
    andwf   RCSTA,W        ; en los bits FERR y OERR
    btfss   STATUS,Z       ;
    goto    Rec_Error      ; Si hay error va a Rec_Error

    btfss   PIR1,RCIF      ; Chequea interrupción por
    retfie  ; recepción serie, si no salimos
    bcf     PIE1,RCIE      ; desactivo la interrupción por recepción

    movf    RCREG,W        ; Se recoge el byte recibido en RCREG
    movwf   TEMP2
    xorlw   A's'           ; y se compara con el código ASCII de la 's'
    btfss   STATUS,Z       ; si no lo es una 's' retornamos de la interrupción
    retfie
    movf    TEMP2,W
    call    SERIAL

INT_ADC
    movlw   0x00           ; Conversión canal AN0
    movwf  ADCH
    call   CONVERSION
    movf   TEMPH,W
    movwf  AD_P1

    movlw   0x01           ; Conversión canal AN1
    movwf  ADCH
    call   CONVERSION
    movf   TEMPH,W
    movwf  AD_P2

    movlw   0x02           ; Conversión canal AN2
    movwf  ADCH
    call   CONVERSION
    movf   TEMPH,W
    movwf  AD_P3

    movf   AD_P1,W        ; Enviamos datos convertidos por serial
    call  SERIAL
    movf   AD_P2,W
    call  SERIAL
    movf   AD_P3,W
    call  SERIAL

    bcf    PIE1,RCIE      ; Reactivamos interrupción por recepción
    retfie

Rec_Error
    ; Llegamos aquí si hubo algún error de recepción
    bcf    RCSTA,4        ; Para limpiar el posible error
    bcf    RCSTA,4        ; limpio CREN y lo vuelvo a poner a 1
    retfie

```

```

;*****
CONVERSION
    call    bank0
    rlf    ADCH,F
    rlf    ADCH,F
    rlf    ADCH,W
    andlw  0x38
    iorlw  0x81
    movwf  ADCON0
    call   TIME25
    bsf    ADCON0,GO

ESPERA
    btfsc  ADCON0,GO
    goto   ESPERA
    movf   ADRESH,W
    movwf  TEMPH
    bcf    ADCON0,0
    return

;*****
; Rutina para envío por puerto serial de la señal diitalizada a 8 bits
;*****
SERIAL
    movwf  TXREG          ; Se carga en TXREG el byte que inicia el envío
    bsf    STATUS,RP0    ; pasamos al banco 1

LAZ01
    btfss  TXSTA,TRMT    ; para poder explorar TXSTA y esperamos a que se transmita
    goto   LAZ01        ; mientras TRMT permanezca a 0 seguimos esperando
    bcf    STATUS,RP0    ; en cuanto TRMT=1 se acabó el envío y volvemos al banco 0
    return              ; Retornamos del subprograma

;*****
TIME25          ;25usec
    MOVWLW  29H
    MOVWF   CONI

T_LP5
    DECFSZ  CONI, F    ;2+3*41-1=124
    GOTO    T_LP5
    RETURN          ;124+1=125*0.2uec

;*****
bank0
    bcf    STATUS,RP0
    bcf    STATUS,RP1
    return

bank1
    bcf    STATUS,RP0
    bcf    STATUS,RP1
    return

;*****

END

```

### 3. Código fuente de la Interfaz Gráfica de Usuario

```
function varargout = SMH2(varargin)
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @SMH2_OpeningFcn, ...
                  'gui_OutputFcn',  @SMH2_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before SMH2 is made visible.
function SMH2_OpeningFcn(hObject, eventdata, handles, varargin)
s = which('SMH2');
[path,name,ext,v]=fileparts(s);
handles.path = path;
handles.data = struct('Cadera',{},'Rodilla',{},'Tobillo',{},'Observacion',{'none'],'t',{});
handles.world = vrworld('male.wrl'); open(handles.world); get(handles.world);
handles.thigh = vrnode(handles.world, 'Bip01_R_Thigh');
handles.calf = vrnode(handles.world, 'Bip01_R_Calf');
handles.foot = vrnode(handles.world, 'Bip01_R_Foot');
handles.veces = 1;
beep on;
cla(handles.axes1);
set(handles.repeticion,'String','1');
set(handles.agregar,'Enable','on');
set(handles.eliminar,'Enable','on');
set(handles.editar,'Enable','on');
set(handles.descargar,'Enable','on');
set(handles.finalizar,'Enable','on');
set(handles.Guardar,'Enable','off');
set(handles.Nombre,'String','');
set(handles.Edad,'String','');
set(handles.Telefono,'String','');
set(handles.panel,'Visible','off');
set(handles.panel_observ,'Visible','off');
set(handles.grafsel,'Enable','off');
set(handles.file_path,'Visible','off');
set(handles.exos,'Value',0);
set(handles.simultaneo,'Enable','off');
set(handles.desfasado,'Enable','off');
set(handles.retardo,'Enable','off');
set(handles.repetitivo,'Enable','off');
set(handles.repeticion,'Enable','off');

if isempty(instrfind)==0
    fclose(instrfind);
end
handles.t = 0;
handles.Rodilla = 0;
handles.Tobillo = 0;
handles.Observ = 0;
```

```

xlabel(handles.axes1,'Tiempo [seg]'); ylabel(handles.axes1,'Ángulo [grados]');
handles.file = open([handles.path '\Data\pacientes.mat']);
handles.npac = length(handles.file.pacientes);

% Creación del puerto serial
try
    handles.port = serial('COM1','BaudRate',2400,'DataBits',8,'StopBits',1,...
        'Parity','none','ReadAsyncMode','Continuous','Timeout',10);
    fopen(handles.port);
    set(handles.comserial,'Checked','On');

catch
    beep
    uiwait(msgbox({'No se ha podido establecer comunicación.',...
        'Reconecte el dispositivo y seleccione el puerto en el menú Comunicación
Serial'},'SERIAL','warn'));
    clc
end

handles.output = hObject;
if ~isfield(handles, 'zoomBtn')
    handles = createToolbarBtns(handles);
end

guidata(hObject, handles);

% ---- Outputs from this function are returned to the command line.
function varargout = SMH2_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;

% -----
function Abrir_Callback(hObject, eventdata, handles)
cd([handles.path '\Data']);
[filename, pathname] = uigetfile( ...
    {'*.mat', 'All MAT-Files (*.mat)'; ...
    '*.*', 'All Files (*.*)'}, ...
    'Abrir sesión registrada');
if isequal([filename,pathname],[0,0])
    return
else
    File = fullfile(pathname,filename);
    Check_And_Load(File,handles)
end

function Check_And_Load(file,handles)
pass = 0;
if exist(file,'file') == 2
    sesion = load(file);
end
campos = fieldnames(sesion);
if (length(campos) == 1) && (strcmp(campos{1},'data'))
    fields = fieldnames(sesion.data);
    if (length(fields) == 5) && (strcmp(fields{1},'Cadera') && strcmp(fields{2},'Rodilla')
&& (strcmp(fields{3},'Tobillo')...
        && (strcmp(fields{4},'Observacion') && (strcmp(fields{5},'t'))
        pass = 1;
    end
end
end

if pass
    handles.LastFile = file;
    handles.Cadera = sesion.data.Cadera;
    handles.Rodilla = sesion.data.Rodilla;
    handles.Tobillo = sesion.data.Tobillo;

```

```

handles.observacion = sesion.data.Observacion;
handles.t = sesion.data.t;
set(handles.file_path, 'Visible', 'on', 'FontName', 'FixedWidth', 'String', handles.LastFile);
set(handles.observacion, 'Enable', 'on', 'String', handles.observacion);
set(handles.panel_observ, 'Visible', 'on');

plot(handles.axes1, handles.t, handles.Cadera, 'g', handles.t, handles.Rodilla, 'r', handles.t, handles.Tobillo, 'b');
set(handles.axes1, 'Color', [0 0 0]);
xlabel(handles.axes1, 'Tiempo [seg]'); ylabel(handles.axes1, 'Angulo [grados]');
ylim([-150 150]);
l=legend('Cadera', 'Rodilla', 'Tobillo', 4);
set(l, 'TextColor', 'w');
grid(handles.axes1, 'on');
set(handles.agregar, 'Enable', 'off');
set(handles.eliminar, 'Enable', 'off');
set(handles.editar, 'Enable', 'off');
set(handles.descargar, 'Enable', 'off');
set(handles.paciente, 'Enable', 'off');
set(handles.grafsel, 'Enable', 'on');
else
errordlg('No es un archivo.mat válido de una sesión registrada', 'Archivo Error')
end
set(handles.paciente, 'Value', 1);
set(handles.Guardar, 'Enable', 'off');
cd(handles.path);
guidata(handles.output, handles);

% --- Executes on button press in descargar.
function descargar_Callback(hObject, eventdata, handles)
estado = get(hObject, 'Value');
switch estado
case 0
set(hObject, 'String', 'Iniciar');
set(handles.exos, 'Enable', 'on');
case 1
set(hObject, 'String', 'Detener');
set(handles.exos, 'Enable', 'off');
val = get(handles.paciente, 'Value');
if val == 1
beep
uiwait(msgbox('Seleccione un Paciente de la lista', 'Warning!!', 'warn'));
set(hObject, 'String', 'Iniciar');
set(handles.exos, 'Enable', 'on');
set(hObject, 'Value', 0);
else
set(handles.agregar, 'Enable', 'off');
set(handles.eliminar, 'Enable', 'off');
set(handles.editar, 'Enable', 'off');
con = get(handles.comserial, 'Checked');
switch con
case 'off'
beep
uiwait(msgbox({'No se ha encontrado el puerto serial COM1', ...
'Habilítelo en el menu Conectar\Comunicación
Serial'}, 'SERIAL', 'warn'));
set(handles.paciente, 'Value', 1);
set(hObject, 'String', 'Iniciar');
set(hObject, 'Value', 0);
clc
case 'on'
fwrite(handles.port, 's');
c=0;
h = waitbar(0, 'Estableciendo comunicación...');

```



```

for i=1:500
    waitbar(i/250)
end
while handles.port.BytesAvailable == 0
    c = c + 1;
    if c == 10
        beep
        close(h);
        uiwait(msgbox({'No se confirmó la petición de conexión. Reconecte el
dispositivo'}, 'SERIAL', 'warn'));
        set(handles.agregar, 'Enable', 'on');
        set(handles.eliminar, 'Enable', 'on');
        set(handles.editar, 'Enable', 'on');
        set(handles.paciente, 'Value', 1);
        set(hObject, 'String', 'Iniciar');
        set(hObject, 'Value', 0);
        return;
    end
end
c=0;
respuesta = fread(handles.port,4);
while respuesta(1) ~= 's'
    c = c + 1;
    if c == 512
        beep
        close(h);
        uiwait(msgbox({'No se confirmó la petición de
conexión'}, 'SERIAL', 'warn'));
        set(handles.agregar, 'Enable', 'on');
        set(handles.eliminar, 'Enable', 'on');
        set(handles.editar, 'Enable', 'on');
        set(handles.paciente, 'Value', 1);
        return
    end
    respuesta = fread(handles.port,4);
end
close(h)
h = waitbar(100, 'OK! - Procesando...');
pause(1)
close(h)
i=1;
t(1)=0;
ang_C(1) = 0;
ang_R(1) = 0;
ang_T(1) = 0;
set(handles.panel_observ, 'Visible', 'off');
cla(handles.axes1)
plot(t, ang_C, 'r', t, ang_R, 'b', t, ang_T, 'g');
set(handles.axes1, 'Color', [0 0 0]);
ylim([-180 180]);
xlabel(handles.axes1, 'Tiempo [seg]'); ylabel(handles.axes1, 'Angulo
[grados]');

l=legend('Cadera', 'Rodilla', 'Tobillo', 4);
set(l, 'TextColor', 'w');
hold on
while estado == 1
    tic
    i=i+1;
    fwrite(handles.port, 's');
    pause(0.1)
    a=dec2bin(fread(handles.port,4),8);
    an0 = bin2dec(a(2,:));
    an1 = bin2dec(a(3,:));
    an2 = bin2dec(a(4,:));
    ang_C(i)=-0.9836*an0+160.98;

```

```

    ang_C(i)=(310/255)*an0-155;
    ang_R(i)=(310/255)*an1-155;
    ang_R(i)=-1.1465*an1+190.7;
    ang_T(i)=(310/255)*an2-155;
    t(i) = t(i-1) + toc;
    %Gráfica
    plot(t,ang_C,'r',t,ang_R,'b',t,ang_T,'g');
    grid
    hold on
    %Motores
    m0 = round(-1.4167*ang_C(i)+127.33);
    m1 = round(-1.4167*ang_R(i)+127.33);
    fwrite(handles.port,[255 0 m0 255 1 255-m1 255 2 127]);
    %Animación 3D
    handles thigh.rotation = [-0.0109*an0+1.7887,0,-0.9251,3.142];
vrdrawnow;
    handles.calf.rotation = [0,1,0,(ang_R(i))*pi/180]; vrdrawnow;
    handles.foot.rotation = [0,1,0,0]; vrdrawnow;
    estado = get(handles.descargar,'Value');
end
fclose(handles.port)
handles.data =
struct('Cadera',ang_C,'Rodilla',ang_R,'Tobillo',ang_T,'Observacion',{'none'},'t',t);
handles.Cadera = handles.data.Cadera;
handles.Rodilla = handles.data.Rodilla;
handles.Tobillo = handles.data.Tobillo;
handles.Observ = handles.data.Observacion;
handles.t = handles.data.t;
set(handles.observacion,'Enable','on','String',handles.Observ);
set(handles.panel_observ,'Visible','on');
set(handles.Guardar,'Enable','on');
set(handles.grafsel,'Enable','on');
set(handles.agregar,'Enable','off');
set(handles.eliminar,'Enable','off');
set(handles.editar,'Enable','off');
fopen(handles.port);
fclose(instrfind);
end
end
set(handles.agregar,'Enable','on');
set(handles.eliminar,'Enable','on');
end
guidata(handles.output,handles);

% --- Executes on button press in finalizar.
function finalizar_Callback(hObject, eventdata, handles)
guardar = get(handles.Guardar,'Enable');
switch guardar
case 'on'
cont = questdlg({'Perderá los datos si no los Guardó previamente'...
'Desea continuar?'],'SMH','Sí','No','No');
switch cont
case 'No'
return
end
end
cla(handles.axes1);
legend off
set(handles.file_path,'Visible','off');
set(handles.panel_observ,'Visible','off');
set(handles.Guardar,'Enable','off');
set(handles.agregar,'Enable','on');
set(handles.eliminar,'Enable','on');
set(handles.editar,'Enable','on');
set(handles.descargar,'Enable','on');

```

```

set(handles.grafsel, 'Enable', 'off');
set(handles.paciente, 'Enable', 'on');
set(handles.paciente, 'Value', 1);
set(handles.grafsel, 'Value', 1);

% --- Executes during object creation, after setting all properties.
function paciente_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject, 'BackgroundColor'), get(0, 'defaultUiControlBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end
s = which('SMH2');
[path, name, ext, v]=fileparts(s);
handles.path = path;
handles.file = open([handles.path '\Data\pacientes.mat']);
pacientes = handles.file.pacientes;
handles.npac = length(pacientes);
paciente{1} = '---Seleccione el PACIENTE---';
for i = 2:handles.npac+1
    paciente{i} = pacientes{i-1}.Nombre;
end
set(hObject, 'String', paciente)

% -----
function Guardar_Callback(hObject, eventdata, handles)
val = get(handles.paciente, 'Value');
str = get(handles.paciente, 'String');
fecha = datestr(now, 'mm-dd-yyyy HH-MM-SS PM');
cd([handles.path '\Data\' str{val}]);
[file, path] = uiputfile('*.mat', 'Guardar como...', fecha);
texto = get(handles.observacion, 'String');
if strcmp(texto, handles.data.Observacion)==0
    handles.data.Observacion = texto;
end
cd(path);
data = handles.data;
save(file, 'data');
cd(handles.path);
guidata(handles.output, handles);

% -----
function Salir_Callback(hObject, eventdata, handles)
cont = questdlg('¿Seguro que Quiere Salir?', 'SMH', 'Si', 'No', 'No');
if isequal(cont, 'Si') == 1
    return
end
switch cont
    case 'No'
        return;
end
try
    fclose(instrfind);
catch
    clc
end
vrclose
delete(instrfind)
delete(handles.output);

% -----
function agregar_Callback(hObject, eventdata, handles)
set(handles.panel, 'Visible', 'on');

% -----
function eliminar_Callback(hObject, eventdata, handles)

```

```

file = open([handles.path '\Data\pacientes.mat']);
pacientes = file.pacientes;
val = get(handles.paciente, 'Value');
str = get(handles.paciente, 'String');
if val == 1
    beep
    uiwait(msgbox('Seleccione un Paciente de la lista', 'Warning!!', 'warn'));
else
    beep
    cont = questdlg({'Esta operación eliminará toda la información del', 'paciente
seleccionado ¿Desea continuar?'}, 'Continuar Eliminación', 'SI', 'No', 'No');
    if isequal(cont, '') == 1
        return
    end
    switch cont
        case 'No'
            set(handles.paciente, 'Value', 1);
            return;
        end
    rmdir([handles.path '\Data\' str{val}], 's');
    val = val-1;
    if val < length(pacientes)
        for i = val:length(pacientes)-1
            pacientes(val) = pacientes(val+1);
        end
        pacientes = pacientes(1:i);
    else
        pacientes = pacientes(1:val-1);
    end
    cd([handles.path '\Data']);
    save('pacientes', 'pacientes');
    cd(handles.path);
    paciente{1} = '---Seleccione el PACIENTE---';
    for i = 2:length(pacientes)+1
        paciente{i} = pacientes{i-1}.Nombre;
    end
    set(handles.paciente, 'String', paciente)
    set(handles.paciente, 'Value', 1);
    msgbox('El paciente ha sido Eliminado', 'BD');
end
guidata(handles.output, handles);

% -----
function comserial_Callback(hObject, eventdata, handles)
estado = get(handles.comserial, 'Checked');
switch estado
    case 'off'
        try
            handles.port = serial('COM1', 'BaudRate', 2400, 'DataBits', 8, 'StopBits', 1, ...
                'Parity', 'none', 'ReadAsyncMode', 'Continuous', 'Timeout', 10);
            fopen(handles.port);
            fwrite(handles.port, 's');
            pause(0.1)
            respuesta = fread(handles.port, 4);
            while respuesta(1) ~= 's'
                c = c + 1;
                if c == 512
                    beep
                    uiwait(msgbox({'No se pudo establecer conexión. Reconecte el
dispositivo'}, 'SERIAL', 'warn'));
                    clc
                    return
                end
            end
            set(handles.comserial, 'Checked', 'On');
end

```

```

        catch
            beep
            uiwait(msgbox({'No se ha encontrado el puerto serial COM1',...
                'Habilítelo en el menu Conectar\Comunicación Serial'}, 'SERIAL', 'warn'));
            clc
        end

    case 'on'
        try
            fclose(instrfind);
        catch
            clc
        end
        set(handles.comserial, 'Checked', 'Off');
    end

end
guidata(handles.output, handles);

% --- Executes on button press in OK.
function OK_Callback(hObject, eventdata, handles)
Nombre = get(handles.Nombre, 'String');
Telefono = get(handles.Telefono, 'String');
Edad = get(handles.Edad, 'String');

if isempty(Nombre)
    beep
    uiwait(msgbox('Debe completar todos los campos', 'Datos del Paciente', 'warn'));
    return
else
    if isempty(Edad)
        beep
        uiwait(msgbox('Debe completar todos los campos', 'Datos del Paciente', 'warn'));
        return
    else
        if isempty(Telefono)
            beep
            uiwait(msgbox('Debe completar todos los campos', 'Datos del Paciente', 'warn'));
            return
        end
    end
end

handles.file = open([handles.path '\Data\pacientes.mat']);
handles.npac = length(handles.file.pacientes);
pacientes = handles.file.pacientes;
i=handles.npac+1;
handles.npac = i;
pacientes{i}.ID = i;
pacientes{i}.Nombre = Nombre;
pacientes{i}.Edad = str2num(Edad);
pacientes{i}.Telefono = Telefono;
s = which('SMH2');
[path, name, ext, v]=fileparts(s);
handles.path = path;
cd([handles.path '\Data']);
save('pacientes', 'pacientes');
cd(handles.path);
set(handles.panel, 'Visible', 'off');
set(handles.eliminar, 'Enable', 'on');

c = length(pacientes);
paciente{1} = '---Seleccione el PACIENTE---';
for i = 2:c+1
    paciente{i} = pacientes{i-1}.Nombre;
end
set(handles.paciente, 'String', paciente)

```

```

set(handles.Nombre,'String','')
set(handles.Edad,'String','')
set(handles.Telefono,'String','')
cd([handles.path '\Data']);
mkdir(Nombre);
cd(handles.path);
msgbox('El paciente ha sido registrado','BD');

% --- Executes on button press in cancel.
function cancel_Callback(hObject, eventdata, handles)
set(handles.panel,'Visible','off');
set(handles.Nombre,'String','')
set(handles.Edad,'String','')
set(handles.Telefono,'String','')

% --- Executes on selection change in grafsel.
function grafsel_Callback(hObject, eventdata, handles)
val = get(handles.grafsel,'Value');
cla(handles.axes1)
switch val
    case 1
plot(handles.axes1,handles.t,handles.Cadera,'r',handles.t,handles.Rodilla,'b',handles.t,hand
les.Tobillo,'g');
        set(handles.axes1,'Color',[0 0 0]);
        xlabel(handles.axes1,'Tiempo [seg]'); ylabel(handles.axes1,'Angulo [grados]');
        ylim([-150 150]);
        l=legend('Cadera','Rodilla','Tobillo',4);
        set(l,'TextColor','w');
        guidata(hObject, handles);
    case 2
plot(handles.axes1,handles.t,handles.Cadera,'r');
        set(handles.axes1,'Color',[0 0 0]);
        xlabel(handles.axes1,'Tiempo [seg]'); ylabel(handles.axes1,'Angulo [grados]');
        ylim([-150 150]);
        grid(handles.axes1,'on');
        guidata(hObject, handles);
    case 3
plot(handles.axes1,handles.t,handles.Rodilla,'b');
        set(handles.axes1,'Color',[0 0 0]);
        xlabel(handles.axes1,'Tiempo [seg]'); ylabel(handles.axes1,'Angulo [grados]');
        ylim([-150 150]);
        grid(handles.axes1,'on');
        guidata(hObject, handles);
    case 4
plot(handles.axes1,handles.t,handles.Tobillo,'g');
        set(handles.axes1,'Color',[0 0 0]);
        xlabel(handles.axes1,'Tiempo [seg]'); ylabel(handles.axes1,'Angulo [grados]');
        ylim([-150 150]);
        grid(handles.axes1,'on');
        guidata(hObject, handles);
end

% -----
function paciente_Callback(hObject, eventdata, handles)
% -----
function Edad_Callback(hObject, eventdata, handles)
% -----
function Telefono_Callback(hObject, eventdata, handles)
% -----
function observacion_Callback(hObject, eventdata, handles)
% -----
function panel_observ_CreateFcn(hObject, eventdata, handles)
% -----
function file_path_CreateFcn(hObject, eventdata, handles)

```

```

% -----
function Archivo_Callback(hObject, eventdata, handles)
% -----
function conexion_Callback(hObject, eventdata, handles)
% -----
function Nombre_Callback(hObject, eventdata, handles)
% -----
function masa_Callback(hObject, eventdata, handles)
% -----
function altura_Callback(hObject, eventdata, handles)
% -----
function C_muslo_Callback(hObject, eventdata, handles)
% -----
function C_pant_Callback(hObject, eventdata, handles)
% -----
function editar_Callback(hObject, eventdata, handles)
% -----

% --- Executes during object creation, after setting all properties.
function Nombre_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
% --- Executes during object creation, after setting all properties.
function Edad_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
% --- Executes during object creation, after setting all properties.
function Telefono_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
% --- Executes during object creation, after setting all properties.
function masa_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
% --- Executes during object creation, after setting all properties.
function altura_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
% --- Executes during object creation, after setting all properties.
function C_muslo_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
% --- Executes during object creation, after setting all properties.
function C_pant_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes during object creation, after setting all properties.
function panel_CreateFcn(hObject, eventdata, handles)
% --- Executes during object creation, after setting all properties.
function grafsel_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
% --- Executes during object creation, after setting all properties.
function observacion_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

```

```

        set(hObject,'BackgroundColor','white');
    end
    % --- Executes during object creation, after setting all properties.
    function figure1_CreateFcn(hObject, eventdata, handles)
    % hObject    handle to figure1 (see GCBO)
    % eventdata  reserved - to be defined in a future version of MATLAB
    % handles    empty - handles not created until after all CreateFcns called

    function handles = createToolBarBtns(handles)

    % load button icons
    load GUIicons

    ht = uitoolbar('Parent', handles.figure1);
    handles.printBtn = uipushtool(ht , ...
        'CData'        , PrintCData, ...
        'ClickedCallback', @print_fcn, ...
        'TooltipString', 'Print Figure');
    handles.zoomBtn = uitoggletool(ht , ...
        'CData'        , ZoomCData, ...
        'ClickedCallback', @zoom_fcn, ...
        'Separator'    , 'on', ...
        'TooltipString', 'Zoom');
    handles.panBtn = uitoggletool(ht , ...
        'CData'        , PanCData, ...
        'ClickedCallback', @pan_fcn, ...
        'TooltipString', 'Pan');
    handles.datatipBtn = uitoggletool(ht , ...
        'CData'        , DataTipCData, ...
        'ClickedCallback', @datatip_fcn, ...
        'TooltipString', 'Data Tip');

    % zoom button callback
    function zoom_fcn(hObject, eventdata) %#ok

    handles = guidata(hObject);
    if strcmpi(get(hObject, 'State'), 'off')
        zoom off;
    else
        set(handles.datatipBtn, 'State', 'off');
        set(handles.panBtn, 'State', 'off');
        datacursormode off;
        pan off;
        zoom on;
    end

    % datatip button callback
    function datatip_fcn(hObject, eventdata) %#ok

    handles = guidata(hObject);
    if strcmpi(get(hObject, 'State'), 'off')
        datacursormode off;
    else
        set(handles.zoomBtn, 'State', 'off');
        set(handles.panBtn, 'State', 'off');
        zoom off;
        pan off;
        datacursormode on;
    end

    % pan button callback

```



```

function pan_fcn(hObject, eventdata) %#ok

handles = guidata(hObject);
if strcmpi(get(hObject, 'State'), 'off')
    pan off;
else
    set(handles.zoomBtn, 'State', 'off');
    set(handles.datatipBtn, 'State', 'off');
    zoom off;
    datacursormode off;
    pan on;
end

% print button callback
function print_fcn(hObject, eventdata, handles) %#ok

handles = guidata(hObject);
printdlg(handles.figure1);

function figure1_ResizeFcn(hObject, eventdata, handles)
% hObject    handle to figure1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes on button press in vrm1.
function vrm1_Callback(hObject, eventdata, handles)
vrplayer_3D = get(handles.vrm1, 'Value');
if vrplayer_3D == 1
    handles.f = vrfigure(handles.world);
    set(handles.f, 'Position', [836 92 432 311]);
else
    vrclose
end

% --- Executes on button press in exos.
function exos_Callback(hObject, eventdata, handles)
if get(handles.exos, 'Value') == 1
    set(handles.simultaneo, 'Enable', 'on');
    set(handles.desfasado, 'Enable', 'on');
    set(handles.retardo, 'Enable', 'on');
    set(handles.repetitivo, 'Enable', 'on');
    set(handles.repeticion, 'Enable', 'on');
    set(handles.agregar, 'Enable', 'off');
    set(handles.eliminar, 'Enable', 'off');
    set(handles.editar, 'Enable', 'off');
    set(handles.descargar, 'Enable', 'off');
    set(handles.finalizar, 'Enable', 'off');
else
    set(handles.simultaneo, 'Enable', 'off');
    set(handles.desfasado, 'Enable', 'off');
    set(handles.retardo, 'Enable', 'off');
    set(handles.repetitivo, 'Enable', 'off');
    set(handles.repeticion, 'Enable', 'off');
    set(handles.agregar, 'Enable', 'on');
    set(handles.eliminar, 'Enable', 'on');
    set(handles.editar, 'Enable', 'on');
    set(handles.descargar, 'Enable', 'on');
    set(handles.finalizar, 'Enable', 'on');
end

% --- Executes on slider movement.
function retardo_Callback(hObject, eventdata, handles)
% hObject    handle to retardo (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB

```

```

% handles    structure with handles and user data (see GUIDATA)

% ---- Executes during object creation, after setting all properties.
function retardo_CreateFcn(hObject, eventdata, handles)
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUITicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

% ---- Executes on button press in simultaneo.
function simultaneo_Callback(hObject, eventdata, handles)
estado = get(hObject,'Value');
if estado == 1
    set(handles.desfasado,'Enable','off');
    set(handles.repetitivo,'Enable','off');
    if isempty(instrfind)==0
        fclose(instrfind);
    end
    fopen(handles.port);
    i = 0;
    while get(handles.simultaneo,'Value') == 1
        i = i+1;
        pause(0.1)
        fwrite(handles.port,'s');
        while handles.port.BytesAvailable < 4
            pause(0.025)
            fwrite(handles.port,'s');
            j=j+1
        end
        a = dec2bin(fread(handles.port,4),8);
        an0(i)=bin2dec(a(2,:));
        an1(i)=bin2dec(a(3,:));
        an2(i)=bin2dec(a(4,:));
        fwrite(handles.port,[255 0 an0(i) 255 1 an1(i) 255 2 an2(i)]);
        an0 = bin2dec(a(2,:));
        an1 = bin2dec(a(3,:));
        an2 = bin2dec(a(4,:));
        ang_C(i)=-0.9836*an0+160.98;
        ang_R(i)=-1.1465*an1+190.7;
        ang_T(i)=(310/255)*an2-155;
        m0 = round(-1.4167*ang_C(i)+127.33);
        m1 = round(-1.4167*ang_R(i)+127.33);
        fwrite(handles.port,[255 0 m0 255 1 255-m1 255 2 127]);
    end
else
    set(handles.desfasado,'Enable','on');
    set(handles.repetitivo,'Enable','on');
end

% ---- Executes on button press in desfasado.
function desfasado_Callback(hObject, eventdata, handles)
estado = get(hObject,'Value');
if estado == 1
    set(handles.simultaneo,'Enable','off');
    set(handles.repetitivo,'Enable','off');
    handles.desfase = round(get(handles.retardo,'Value'));
    j = handles.desfase/0.1;
    k = 0;
    i = 0;
    if isempty(instrfind)==0
        fclose(instrfind);
    end
    fopen(handles.port);
    while get(handles.desfasado,'Value') == 1
        i = i+1;

```

```

        pause(0.1)
        fwrite(handles.port,'s');
        while handles.port.BytesAvailable < 4
            pause(0.025)
            fwrite(handles.port,'s');
        %     j=j+1
        end
        a = dec2bin(fread(handles.port,4),8);
        an0 = bin2dec(a(2,:));
        an1 = bin2dec(a(3,:));
        an2 = bin2dec(a(4,:));
        ang_C(i)=-0.9836*an0+160.98;
        ang_R(i)=-1.1465*an1+190.7;
        ang_T(i)=(310/255)*an2-155;
        m0(i) = round(-1.4167*ang_C(i)+127.33);
        m1(i) = round(-1.4167*ang_R(i)+127.33);
        if i > j
            fwrite(handles.port,[255 0 m0(i-j) 255 1 255-m1(i-j) 255 2 127]);
        end
    end
else
    set(handles.simultaneo,'Enable','on');
    set(handles.repetitivo,'Enable','on');
end

% --- Executes on button press in repetitivo.
function repetitivo_Callback(hObject, eventdata, handles)
estado = get(hObject,'Value');
if estado == 1
    set(handles.simultaneo,'Enable','off');
    set(handles.desfasado,'Enable','off');
    if isempty(instrfind)==0
        fclose(instrfind);
    end
    fopen(handles.port);
    for i=1:handles.veces
        for j=1:length(handles.Cadera)
            pause(0.2)
            fwrite(handles.port,[255 0 (-1.4167*handles.Cadera(j)+127.33) 255 1
(255+1.4167*handles.Rodilla(j)-127.33) 255 2 127]);
        end
    end
else
    set(handles.simultaneo,'Enable','on');
    set(handles.repetitivo,'Enable','on');
end

function repeticion_Callback(hObject, eventdata, handles)
handles.veces = round(str2double(get(hObject,'String')));
if isnan(handles.veces)
    errordlg('You must enter a numeric value','Bad Input','modal')
    return
end
guidata(hObject,handles);

% --- Executes during object creation, after setting all properties.
function repeticion_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUIControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

# GUIA DE USUARIO PARA EL MODELO EXPERIMENTAL

A continuación se señalan brevemente los pasos a seguir para poder emplear el sistema LEGXOS y su plataforma bípida experimental.

## Pasos para conectar e iniciar el sistema:

1. Conectar la unidad de adquisición de datos al puerto serial del computador, empleando para ello un cable serial directo con conectores DB9 hembra-macho. Es importante asegurarse de que el puerto serial utilizado sea el COM1.
2. Acoplar el exoesqueleto pasivo a la pierna izquierda de la persona. Asegurarse de que las articulaciones de la estructura estén alineadas con la cadera y rodilla de la persona. Si el sistema está bien ajustado, no debe existir incomodidad al realizar movimientos con la pierna.
3. Energizar la unidad de adquisición y la plataforma bípida. Asegurarse de que esté conectado el adaptador de 9V a la unidad de adquisición, y que el bípido tenga las baterías o esté alimentado a una fuente externa.
4. Iniciar la aplicación LEGXOS en el computador. Al abrir la ventana del programa no debe presentarse mensaje de error. Si aparece un error se debe al puerto serial. Siga las instrucciones del mensaje y verifique que ninguna otra aplicación esté haciendo uso del puerto serial COM1.

## Para agregar paciente:

1. Haga click en el botón *Agregar*.
2. Aparecerá un formulario, complete todos los campos.
3. Si desea registrar los datos ingresados haga click en el botón *Registrar*, de lo contrario haga click en *Cancelar*.
4. Luego de agregar un paciente, en el directorio raíz de la aplicación se creará una carpeta con el nombre de dicho paciente, en la cual se almacenarán las observaciones y datos de cada sesión de terapia.

## Para eliminar paciente:

1. Seleccione el paciente que desea eliminar del menú desplegable que contiene la lista de personas registradas en el sistema.
2. Haga click en el botón *Eliminar*.
3. Aparecerá un mensaje de alerta indicando que se eliminará toda la información relacionada con ese paciente.
4. Si desea eliminar el paciente haga click en OK, de lo contrario cancele la operación.

**Para realizar una sesión de sólo monitoreo:**

1. Seleccione el paciente deseado del menú desplegable que contiene la lista de personas registradas en el sistema.
2. Haga click en *Iniciar captura*.
3. Aparecerá una barra de proceso mientras se verifica la comunicación entre la unidad de adquisición y el computador. Si todo está en orden se iniciará la captura del movimiento ejecutado con el exoesqueleto pasivo, de lo contrario aparecerá un mensaje de fallo en la comunicación, siga las instrucciones que se le den en pantalla.
4. Haga click en *Detener captura*, cuando desee finalizar el monitoreo del movimiento.
5. Complete el campo de *Observaciones* si desea agregar algunas al movimiento analizado.
6. Haga click en *Finalizar sesión*. Se notificará que perderá los datos si no guardó previamente la sesión. Siga las instrucciones que se den en pantalla.

**Para guardar una sesión:**

1. Luego de haber ejecutado una sesión de captura de movimiento, haga click en el menú *Archivo -> Guardar*.
2. Aparecerá una ventana del sistema Windows para guardar en un directorio dado. Haga click en *Aceptar* o *Guardar*. Si desea puede cambiar el directorio y el nombre predeterminado del archivo, aunque no se recomienda para mantener un orden, ya que el sistema lo guarda automáticamente con la fecha y la hora de la prueba y lo ubica en la carpeta del paciente seleccionado.

**Para abrir una sesión registrada en el sistema:**

1. Haga click en el menú *Archivo -> Abrir*.
2. Seleccione la carpeta del paciente deseado.
3. Seleccione el archivo de sesión deseado.
4. Haga click en *Abrir* y en pantalla se graficará el movimiento guardado y se cargarán los datos en el sistema para poder emplearlos en la plataforma bípeda en modo repetitivo si se desea.

**Para visualizar el movimiento en 3D:**

1. Inicie captura.
2. Seleccione la opción denominada *Animación 3D*, dentro del submenú *Visualizaciones*.
3. Si desea cerrar la ventana del entorno gráfico, elimine la marca de selección.

**Para realizar una sesión de captura y accionamiento de la plataforma bípeda:**

1. Luego de iniciar el sistema, seleccione la opción denominada *Plataforma bípeda*, dentro del submenú *Visualizaciones*.
2. Seleccione el modo de operación deseado, ya sea *Simultáneo*, *Desfasado* o *Repetitivo*.
3. El modo simultáneo ejecutará el accionamiento a la par que recibe los datos.
4. El modo desfasado ejecutará el movimiento con un retardo temporal seleccionado en la barra de desplazamiento de dicho modo de operación.
5. En el modo repetitivo debe haberse capturado previamente un movimiento o abierto un archivo de datos previamente guardado. Este modo ejecutará ese patrón de movimiento la cantidad de veces que se indique en el campo de texto para el número de repeticiones deseadas.
6. Para detener el accionamiento simplemente elimine la marca de selección de la plataforma bípeda.

**NOTAS:**

- Al encender tanto la unidad de adquisición como la plataforma bípeda, se deben observar encendidos los LEDs indicadores de cada sistema. Si no hay fallos en la comunicación durante el uso de la plataforma bípeda, la tarjeta controladora de servomotores debe presentar parpadeo en su LED indicador.
- En caso de presentarse problemas de comunicación, asegúrese que haya conectado el puerto serial COM1 y que no sea empleado por otra aplicación.
- En caso de escuchar un sonido durante la captura de movimiento y observa que se detiene el monitoreo, reinicie la captura. Si aparece mensaje de fallo en la comunicación, presione el pulsador de reset en la unidad de adquisición. Si el problema continúa, haga click en *Archivo->Salir* y reinicie la aplicación.
- En caso de observar que la plataforma bípeda no es accionada, verifique la comunicación. Si ésta falla presione el reset de la unidad de adquisición e intente de nuevo.
- Recuerde mantener cargadas las baterías de la plataforma bípeda o emplear una fuente externa de alimentación de 5V capaz de otorgar mínimo dos amperios.