



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA
Y MECÁNICA**

CARRERA DE INGENIERÍA MECATRÓNICA

**TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL
TÍTULO DE INGENIERO EN MECATRÓNICA**

**TEMA: DISEÑO Y CONSTRUCCIÓN DE UN TORSO
ROBÓTICO COLABORATIVO DE DOS BRAZOS PARA LA
UNIÓN DE UNA JUNTA EMPERNADA EN UNA CELDA DE
TRABAJO**

**AUTORES: CARVAJAL MORALES, ANAHÍ SALOMÉ
MARTÍNEZ PEÑAHERRERA, SONIA SALOMÉ**

DIRECTOR: MSc. LOZA MATOVELLE, DAVID CÉSAR

**SANGOLQUÍ
2016**



DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y MECÁNICA
CARRERA DE INGENIERÍA MECATRÓNICA

CERTIFICACIÓN.

Certifico que el trabajo de titulación, *“DISEÑO Y CONSTRUCCIÓN DE UN TORSO ROBÓTICO COLABORATIVO DE DOS BRAZOS PARA LA UNIÓN DE UNA JUNTA EMPERNADA EN UNA CELDA DE TRABAJO”*. Realizado por las señoritas **CARVAJAL MORALES ANAHÍ SALOMÉ** y **MARTÍNEZ PEÑAHERRERA SONIA SALOMÉ**, ha sido revisado en su totalidad, el mismo que cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de las Fuerzas Armadas ESPE, por lo tanto, me permito acreditarlo y autorizar a las señoritas **CARVAJAL MORALES ANAHÍ SALOMÉ** y **MARTÍNEZ PEÑAHERRERA SONIA SALOMÉ** para que lo sustenten públicamente.

Sangolquí, 28 de octubre del 2016

Atentamente,



MSc. LOZA MATOVELLE, DAVID CÉSAR



DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y MECÁNICA
CARRERA DE INGENIERÍA MECATRÓNICA

AUTORÍA DE RESPONSABILIDAD

Nosotras, **CARVAJAL MORALES ANAHÍ SALOMÉ** y **MARTÍNEZ PEÑAHERRERA SONIA SALOMÉ**, con cédula de identidad N° 100315867-0 y 172098915-9 respectivamente, declaramos que éste trabajo de titulación ***“DISEÑO Y CONSTRUCCIÓN DE UN TORSO ROBÓTICO COLABORATIVO DE DOS BRAZOS PARA LA UNIÓN DE UNA JUNTA EMPERNADA EN UNA CELDA DE TRABAJO.”*** Ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaramos que éste trabajo es de nuestra autoría, en virtud de ello nos declaramos responsables del contenido, veracidad y alcance de la investigación mencionada.

Sangolquí, 28 de octubre del 2016

Atentamente:

Carvajal Morales

Anahí Salomé

C.C. 100315867-0

Martínez Peñaherrera

Sonia Salomé

C.C. 172098915-9



DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y MECÁNICA
CARRERA DE INGENIERÍA MECATRÓNICA

AUTORIZACIÓN

Nosotras, **CARVAJAL MORALES ANAHÍ SALOMÉ** y **MARTÍNEZ PEÑAHERRERA SONIA SALOMÉ**, autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar en la biblioteca Virtual de la institución el presente trabajo de titulación ***“DISEÑO Y CONSTRUCCIÓN DE UN TORSO ROBÓTICO COLABORATIVO DE DOS BRAZOS PARA LA UNIÓN DE UNA JUNTA EMPERNADA EN UNA CELDA DE TRABAJO.”*** Cuyo contenido, ideas y criterios son de nuestra autoría y responsabilidad.

Sangolquí, 28 de octubre del 2016

Atentamente:

Carvajal Morales
Anahí Salomé

C.C. 100315867-0

Martínez Peñaherrera
Sonia Salomé

C.C. 172098915-9

Dedicatoria

Este trabajo de titulación lo dedicamos a nuestras familias, quienes forjaron nuestras vidas desde un inicio, guiándonos en principios de convivencia y orientándonos a romper barreras para que como mujeres podamos adquirir una profesión encaminada a la producción en servicio a los demás.

Anahí y Sonia

Agradecimientos

A nuestros padres, para quienes somos sus niñas bonitas y que con alegría y firmeza nos han sabido encaminar al cumplimiento de nuestras metas. A nuestras madres, quienes además de darnos su ejemplo con ternura, nos enseñaron a luchar para lograr nuestros objetivos. A nuestros hermanos, han sido nuestro ejemplo y apoyo.

A los maestros de toda la vida, con los que aprendimos, desde leer y escribir hasta desarrollar tecnología e investigación. Sobre todo a aquellos quienes supieron exigirnos para cumplir nuestras metas y seguir nuestros sueños.

A nuestro tutor, MSc. David Loza M., quien nos encaminó durante todo el proyecto y supo guiarnos hacia el convencimiento de que nada es imposible y que el campo de la investigación es un universo entero que involucra un sin fin de conocimientos.

Como no agradecer al MSc. Alexander Ibarra, al Ing. Roberty Velasco, al Sr. Nelson Cárdenas, a los docentes del DECEM y los responsables del laboratorio de Manufactura, quienes estuvieron siempre prestos a orientarnos con su conocimiento y experiencia e hicieron amenas las largas horas de trabajo.

A nuestros compañeros tesisistas, con los que compartimos experiencias, conocimientos, trabajo, solidaridad, alegrías y frustraciones. Aquellos quienes estuvieron siempre dispuestos a darnos ánimo y brindarnos su apoyo y confianza, haciendo que se solidifiquen esos lazos de amistad que duran para toda la vida.

Como no agradecer además a Cristian, Carlos, Freddy y Santiago, amigos incondicionales de la carrera, con quienes hemos compartido trabajos, proyectos, desvelos, pero sobre todo alegrías, sueños y proyecciones. Personas que han significado un gran respaldo y apoyo permanente durante esta etapa de nuestras vidas.

Resumen

El presente trabajo corresponde al diseño y construcción de un torso robótico colaborativo de dos brazos para la unión de una junta empernada en una celda de trabajo. El objetivo principal del proyecto es la implementación de un manipulador colaborativo capaz de resolver un proceso automático mediante el uso de software libre y la incorporación de visión artificial. A través de la construcción de un segundo brazo antropomorfo basado en el proyecto de titulación elaborado por Sánchez y Terán, el diseño e implementación del torso del manipulador y la celda de trabajo, se obtuvo la estructura del proyecto. Se implementó un planificador para realizar el control de cada uno de los brazos mediante máquina de estados, adicionalmente se creó la interfaz gráfica del programa para manipular el movimiento manual y automático del torso robótico.

PALABRAS CLAVES:

- **TORSO ROBÓTICO COLABORATIVO**
- **VISIÓN ARTIFICIAL**
- **ALGORITMOS GENÉTICOS**
- **PROGRAMACIÓN OFFLINE**
- **MANIPULADOR ROBÓTICO REDUNDANTE**

Abstract

This work corresponds to the design and construction of a collaborative robotic torso with two arms for attachment of a bolted joint in a workcell. The main objective of the project is the implementation of a collaborative manipulator capable of solving an automatic process by using open source and incorporating artificial vision. The project structure was built by the construction of a second anthropomorphic arm based on the research by Sánchez y Terán, the design and implementation of the torso of the manipulator and the workcell. It was implemented a scheduler to control each of the arms by state machine. Additionally, it was created a graphical interface program to manipulate the manual and automatic movement of the robotic torso.

KEYWORDS:

- **COLLABORATIVE ROBOTIC TORSO**
- **ARTIFICIAL VISION**
- **GENETIC ALGORITHMS**
- **OFFLINE PROGRAMMING**
- **ROBOTIC REDUNDANT MANIPULATOR**

Índice general

Certificado	ii
Autoría de responsabilidad	iii
Autorización	iv
Dedicatoria	v
Agradecimientos	vi
Resumen	vii
Abstract	viii
Índice de figuras	xiii
Índice de tablas	xvii
1. Generalidades	2
1.1. Antecedentes	2
1.2. Justificación e Importancia	4
1.3. Alcance	5
1.4. Objetivos	5
1.4.1. Objetivo General	5
1.4.2. Objetivos Específicos	5
1.5. Estructura del Documento	6
2. Estado del Arte	7
2.1. Manipuladores Robóticos Industriales	7
2.1.1. Configuración Cartesiana	7
2.1.2. Configuración Cilíndrica	8
2.1.3. Configuración Polar	8

2.1.4.	Configuración Angular	9
2.1.5.	Configuración SCARA	9
2.2.	Manipuladores Colaborativos	10
2.3.	Selección de Manipuladores y Aplicación en la Industria	12
2.4.	Efectores Finales	13
2.5.	Celda de Trabajo	15
2.6.	Cinemática de Manipuladores Robóticos	16
2.6.1.	Cinemática Directa	16
2.6.1.1.	Aspectos Importantes en la Solución de un Problema Cinemático	17
2.6.1.2.	Resolución del Algoritmo de Denavit Hartenberg	17
2.6.2.	Cinemática Inversa	19
2.6.2.1.	Métodos Geométricos	20
2.6.2.2.	Desacoplamiento Cinemático	20
2.6.2.3.	Algoritmos Genéticos	20
2.7.	Generación y Optimización de Trayectoria para Robots Colaborativos	22
2.8.	Técnicas de Control para Robots Colaborativos	23
2.8.1.	Planificador y Control Jerárquico	24
2.9.	Sensado y Sistema de Visión Artificial	25
2.9.1.	Detección de Geometría	28
2.10.	Manipulador Antropomorfo de Siete Grados de Libertad	31
2.10.1.	Características del Manipulador Antropomórfico de Siete Grados de Libertad	32
2.10.2.	Análisis Cinemático y Dinámico mediante Algoritmos Genéticos	32
2.10.2.1.	Parámetros de Denavit Hartenberg	33
2.10.2.2.	Solución de Cinemática Inversa mediante Algoritmo Genético	34
2.10.2.3.	Generación de Trayectorias	35
2.10.2.4.	Resultados	35
2.11.	Resumen	36
3.	Metodología y Diseño Mecatrónico	37
3.1.	Metodología de Diseño	37
3.1.1.	Corriente de Especificación	38
3.1.1.1.	Conceptos de Operaciones	38
3.1.1.2.	Requisitos y Arquitectura del Sistema	38
3.1.1.3.	Diseño Detallado	38
3.1.2.	Corriente de Desarrollo	39

3.1.3.	Corriente de Pruebas	39
3.1.4.	Despliegue de la Función de Calidad QFD	39
3.1.5.	Factibilidad y Planeación	42
3.2.	Módulo 1. Estructura	43
3.2.1.	Subsistema 1.1. Brazos Antropomórficos	43
3.2.1.1.	Diseño Geométrico	43
3.2.1.2.	Movilidad de los Eslabones	50
3.2.1.3.	Esquema de Conexión	53
3.2.1.4.	Configuración de los Motores Dynamixel	53
3.2.1.5.	Manipulación de los Motores Dynamixel desde Ubuntu	55
3.2.2.	Subsistema 1.2. Torso Robótico	57
3.2.2.1.	Desarrollo del Diseño	59
3.2.2.2.	Análisis Estático	61
3.2.3.	Subsistema 1.3. Modificación del Gripper	70
3.3.	Módulo 2. Cinemática del Manipulador	71
3.3.1.	Subsistema 2.1. Cinemática Inversa en ROS	71
3.4.	Módulo 3. Visión Artificial	73
3.4.1.	Subsistema 3.1. Sistema de Detección de Objetos	73
3.5.	Módulo 4. Planificador	74
3.5.1.	Subsistema 4.1. Interfaz Gráfica	74
3.5.1.1.	Modo Manual	75
3.5.1.2.	Modo Automático	76
3.5.1.3.	Botón Parar	76
3.5.1.4.	Mensajes de Información	76
3.5.1.5.	Mensajes de información de estado del proceso	77
3.5.1.6.	Mensajes de Alerta	78
3.5.2.	Subsistema 4.2. Máquina de Estados	78
3.6.	Resumen	80
4.	Implementación, Pruebas y Resultados	81
4.1.	Implementación	81
4.1.1.	Estructura	81
4.1.2.	Implementación Acople para Cámara	87
4.1.3.	Panel de Control	88
4.1.4.	Placa PCB	89
4.2.	Pruebas y Resultados	89

4.2.1.	Primera prueba, funcionamiento de los componentes instalados en el sistema	90
4.2.2.	Segunda prueba, funcionamiento del programa de algoritmos genéticos para la resolución de la cinemática inversa de los brazos	91
4.2.3.	Tercera prueba, sistema de detección de objetos por medio de visión artificial	92
4.2.4.	Cuarta prueba, comunicación entre nodos del sistema	94
4.2.5.	Quinta prueba, funcionamiento general del sistema	96
4.3.	Resumen	104
5.	Conclusiones, Recomendaciones y Trabajos Futuros	105
5.1.	Conclusiones	105
5.2.	Recomendaciones y Trabajos Futuros	107
	Bibliografía	109

Índice de figuras

1.	Proceso automatizado en BJ-Gear.	3
2.	Robot Cartesiano	8
3.	Configuración Cilíndrica.	8
4.	Configuración Polar.	9
5.	Configuración Angular.	9
6.	Configuración SCARA.	10
7.	YuMi - robot colaborativo de dos brazos	11
8.	Robot dual colaborativo de dos brazos	11
9.	Diagrama de aplicaciones comunes de MRI en la industria.	12
10.	Aplicaciones para a) Robot FANUC soldador, b) Robot KUKA en un sistema de paletización	13
11.	Clasificación de efectores finales.	13
12.	Tipos de agarre humano a) cilíndrico b) entre los extremos c) gancho d) palmar e) esférico f) lateral.	14
13.	Configuraciones de agarre a) planar b) espacial c) palmar d) agarre con carga.	14
14.	Zona de trabajo cooperativo del manipulador	15
15.	Layout de un robot colaborativo de dos brazos	15
16.	Problema cinemático.	16
17.	Problema cinemático en manipuladores, representación PUMA 260	16
18.	Ejes convencionales para uso de algoritmo de Denavit Hartenberg	18
19.	Control punto a punto.	23
20.	Control por trayectoria.	23
21.	Robot Kawada Nextage Open de Tecnalía. Robot bi-brazo	25
22.	Digitalización de imágenes.	26
23.	Uso de tecnologías CCD y CMOS.	27
24.	Procesamiento y análisis de imágenes	27
25.	Detección de imágenes y proceso de digitalización	28

26.	Detección de formas a) Imagen original b) Imagen convertida a binaria c) Imagen filtrada para detección de bordes	29
27.	Ejemplo de elementos estructurales típicos	29
28.	Diagrama de flujo para reconocimiento por forma	30
29.	Imágenes a) original b) dilatada c) erosionada	30
30.	Configuración robot redundante.	31
31.	Vista Isométrica realizada esquemáticamente con RoboAnalyzer, del robot antropomórfico de 7 GDL.	32
32.	Diagrama que representa los parámetros de Denavit Hartenberg.	33
34.	Metodología de desarrollo del torso - Método en V	38
35.	Casa de la calidad para el torso robótico colaborativo	41
36.	Base del manipulador	44
37.	Eslabón 1	44
38.	Eslabón 2	45
39.	Eslabón 3	46
40.	Eslabón 4	47
41.	Eslabón 5	47
42.	Eslabón 6	48
43.	Soporte de Gripper	49
44.	Efector final tipo gripper	49
45.	Geometría del robot manipulador	50
46.	Control de Dynamixel desde PC	52
47.	Puertos de comunicación del Controlador USB2Dynamixel	53
48.	Esquema de conexión Torso Robótico	53
49.	Diagrama de configuración de motores en Wizard	54
50.	Búsqueda e identificación de los motores conectados en serie	55
51.	Paquetes de instalación de PyPot y programas adicionales	55
52.	Paquetes requeridos instalados en Ubuntu	56
53.	Ejecución del programa Herborist desde el Terminal	56
54.	Espacio de trabajo del programa Herborist	56
55.	Diseño 1	57
56.	Diseño 2	58
57.	Diseño de la base del torso	59
58.	Torso	59
59.	Tapas del torso	60
60.	Estructura mecánica del Torso robótico	60
61.	Diseño final de manipulador colaborativo de dos brazos	61

62.	Análisis de volcamiento	62
63.	Dimensiones para el cálculo de aplastamiento en agujeros de la base del Manipulador.	66
64.	Acople para cámara	70
65.	Cinemática inversa	71
66.	Algoritmos genéticos	72
67.	Diagrama de flujo, Visión artificial	74
68.	HMI	75
69.	Modo Manual	75
70.	Modo Automático	76
71.	Botón Parar	76
72.	Mensaje de información: a) calculo inicializado, b) calculo terminado, c) posición ejecutada	77
73.	Mensaje de información: a) torso en posición Home, b) secuencia inicializada, c) secuencia terminada	77
74.	Mensajes de información del estado del proceso: a) Ubicando, b) Ubicado	78
75.	Mensajes de alerta	78
76.	Diagrama de flujo del sistema	80
76.	Configuración para la impresión 3D de los eslabones del segundo brazo del manipulador colaborativo	82
77.	Impresión 3D de los eslabones del segundo brazo del manipulador colaborativo	82
78.	Procesos de a) corte b) rolado c) perforación d) trazado y doblado e) soldadura f) masillado g) pintura	83
79.	Implementación del torso robótico colaborativo de dos brazos antropomorfos completo	87
80.	Acople cámara implementación	87
81.	Panel de Control	88
82.	Placa del circuito electrónico	89
83.	Conexión de motores desde el terminal	91
84.	Resolución de cinemática inversa mediante algoritmos genéticos – posición inaccesible a) algoritmo sin límites b) algoritmo con límites (caso real)	92
85.	Resolución de cinemática inversa mediante algoritmos genéticos – posición accesible a) algoritmo sin límites b) algoritmo con límites (caso real)	92

86.	Resultado de filtro con rangos iniciales	93
87.	Resultado de filtro con rangos finales	93
88.	Tópicos del sistema Torso Robótico	94
89.	Visualización de la comunicación entre nodos del sistema Torso Robótico	95
90.	HMI Torso Robótico	97
91.	Información, inicio de cálculo de cinemática	97
92.	Información, cálculo finalizado	98
93.	Información, posición ejecutada	98
94.	Información de posicionamiento, ubicando	99
95.	Información de posicionamiento, ubicado	99
96.	Posicionamiento de cada estado	99
97.	Información, torso en Home	100
98.	Funcionamiento Home	100
99.	Información, secuencia inicializada	100
100.	Funcionamiento secuencia primera parte	101
101.	Activación del sistema de detección de elementos	101
102.	Mensaje de alerta 1	102
103.	Mensaje de alerta 2	102
104.	Funcionamiento de la secuencia segunda parte	103
105.	Información, secuencia terminada	103
106.	Funcionamiento Parar	103

Índice de tablas

1.	Tipos de robots industriales según ISO8373	10
2.	Parámetros de Denavit Hartenberg del robot AR7D	34
3.	Restricciones de las articulaciones del robot	35
4.	Exactitud del robot ARD7	35
5.	Repetibilidad del robot ARD7	35
6.	Necesidades del cliente de la QFD	40
7.	Características técnicas de la QFD	40
8.	Resultados de la matriz QFD	42
9.	Estructura modular del Torso robótico	43
10.	Análisis estático de la Base del manipulador	44
11.	Análisis estático de la base del Eslabón 1	45
12.	Análisis estático de la base del Eslabón 2	45
13.	Análisis estático del Eslabón 3	46
14.	Análisis estático del Eslabón 4	47
15.	Análisis estático del Eslabón 5	48
16.	Análisis estático del Eslabón 6	48
17.	Análisis estático del Soporte del Gripper	49
18.	Motores Dynamixel	51
19.	Corriente necesaria para los motores Dynamixel de los manipuladores	51
20.	Tabla comparativa de diseños preliminares	58
21.	Análisis de Tensiones en las tapas del Torso	64
22.	Análisis de desplazamiento en las tapas del Torso	64
23.	Análisis de deformaciones unitarias en las tapas del Torso	65
24.	Análisis de Factor de Seguridad en las tapas del Torso	65
25.	Análisis de Tensiones de la base del Manipulador	68
26.	Análisis de desplazamiento de la base del Manipulador	68
27.	Análisis de deformaciones unitarias de la base del Manipulador	69
28.	Análisis de Factor de Seguridad de la base del Manipulador	69

29.	Parámetros de impresión	82
30.	Hoja de Procesos, Base del Torso	84
31.	Hoja de Procesos, Torso	85
32.	Hoja de Procesos, Tapas laterales del Torso	86
33.	Corriente y Voltaje de los componentes instalados	90
34.	Rangos iniciales de color verde	93
35.	Rangos finales de color verde	93
36.	Nodos ROS del torso robótico	96

Introducción

En la actualidad a nivel industrial se hace uso de manipuladores robóticos de diferentes configuraciones, para desarrollar actividades repetitivas que representan peligro para los operarios. En los últimos años se ha incrementado la implementación de procesos automatizados en empresas importantes como por ejemplo en Johnson & Johnson con maquinaria desarrollada por Universal Robots [Robots, 2016].

En algunos procesos industriales se observó la necesidad de incluir manipuladores que realicen trabajos colaborativos, como solución a esta problemática se realizó la inclusión de torsos robóticos colaborativos como IREX 2013 [EPSON, 2016] y SDA20D [MOTOMAN, 2016]. Estos robots manipuladores desarrollados por EPSON y YASKAWA respectivamente están equipados con funciones de visión artificial y diversos sensores; aplican la más alta tecnología y cumplen con las normas de calidad y seguridad competentes para las aplicaciones de montaje y manejo de material.

El proyecto propone la construcción de un torso robótico colaborativo de dos brazos, donde cada uno consta de elementos y geometría precisa para realizar una actividad colaborativa en una celda de trabajo. En este proyecto se pretende realizar el control y programación de un torso para ejecutar un proceso automático para empernar una junta como aplicación al desarrollo.

Para el desarrollo del presente proyecto se partió del estudio previo ejecutado como proyecto de investigación en la Universidad de las Fuerzas Armadas ESPE en el cual se realizó el diseño y construcción de un brazo robótico de 7 grados de libertad, con análisis cinemático y dinámico mediante algoritmos genéticos [Sánchez & Terán, 2015]. Se realizó la construcción de un segundo brazo antropomorfo de las mismas características dispuestas en el trabajo mencionado, con la inclusión de los dos brazos antropomorfos del manipulador acoplados al torso; de esta manera se puede realizar varias actividades y diversos procesos colaborativos.

Capítulo 1

Generalidades

En el presente capítulo se detallan los antecedentes, justificación, alcance y objetivos del proyecto de investigación desarrollado. Durante el capítulo se menciona la importancia del estudio e inclusión de manipuladores colaborativos dentro de la industria y se hace referencia a varias investigaciones relacionadas realizadas a nivel mundial, nacional e institucional. Por último, se realiza una breve descripción del contenido de los siguientes capítulos.

1.1. Antecedentes

A nivel mundial, el uso de manipuladores robóticos en la industria ha sido uno de los avances tecnológicos más importantes en cuanto a automatización de procesos. Según Metalactual [Correa et al., 2011], los manipuladores robóticos son dispositivos que han ganado aceptación en el medio industrial gracias a su capacidad de llevar a cabo un número variado de tareas con rapidez y precisión. Además, se pueden adaptar a diversas situaciones con sólo modificar el programa que comanda los movimientos de la máquina. El incremento de su uso también obedece a que permiten reducir costos, mejorar la calidad del producto, aumentar la producción, reducir desperdicios, mejorar la seguridad del puesto de trabajo y ahorrar espacio.

En países de Europa existe un avance tecnológico a tal magnitud que han logrado reducir al máximo los costos de producción en sus productos y al mismo tiempo aumentado la calidad de los mismos. Un ejemplo de este avance es la empresa Universal Robots, quienes se dedican a dar soluciones industriales mediante la inclusión de robots manipuladores. Uno de los casos prácticos de automatización fue la incorporación de cuatro brazos robóticos a la plantilla de BJ-Gear [Robots, 2016]. Cada manipulador se encarga de realizar un proceso diferente como pintado de engranajes, desbarbado, estampado y terminado de piezas como se muestra en la Figura 1.



Figura 1: Proceso automatizado en BJ-Gear.

Fuente: [Robots, 2016]

En ABB [ABB, 2016] se menciona que pocos escenarios de producción están cambiando tan rápidamente como el montaje de piezas pequeñas. La industria de la electrónica, en particular, ha visto dispararse la demanda de productos más allá de la oferta de mano de obra especializada. Como métodos de montaje convencionales disminuyen de valor, los fabricantes están encontrando estratégica y económicamente imperativo invertir en nuevas soluciones. Una de las soluciones a problemas de montaje rápido y el aumento del ritmo de introducción de nuevos productos ha sido la inclusión de robots duales a la industria. Dentro de los robots de configuración dual, de doble brazo, existen alrededor del mundo varias empresas dedicadas a la creación de manipuladores robóticos de este tipo, una de ellas es la empresa EPSON. Esta empresa creó IREX 2013 [EPSON, 2016], equipado con funciones de visión y la fuerza con sensores, este robot puede ejecutar de forma autónoma tareas mediante el reconocimiento de objetos, la toma de decisiones, y el ajuste de la cantidad de fuerza aplicada, sobre la marcha. Así también tenemos a YASKAWA, quienes desarrollaron el robot SDA20D [MOTOMAN, 2016], un robot de doble brazo de 15 ejes. SDA20D fue diseñado para el montaje y material complejo aplicaciones de manipulación.

En el país ha existido un avance tecnológico importante debido al interés de varias personas en temas de robótica industrial y a la necesidad de optimización de tiempo, recursos y liberación de personal designado a tareas repetitivas y de alta precisión en diferentes empresas. Uno de estos casos es el de la empresa Soft Tetratonic conformada por tres jóvenes ecuatorianos [COMERCIO, 2010]. A pesar de ser una empresa nueva en el mercado, se convirtieron en distribuidores a nivel nacional de la firma alemana KUKA, unas de las más reconocidas en robótica industrial. La empresa recibe los brazos robóticos estándar y adapta un efector final específico de acuerdo a las necesidades del cliente.

En la Universidad de las Fuerzas Armadas – ESPE, se ha dado lugar al desarrollo

de varios proyectos que contemplan el diseño y construcción de procesos automatizados. Se ha impulsado el uso de software y hardware libre que contribuyen a la educación y desarrollo de tecnología de bajo costo que implica el desarrollo de prototipos prometedores. Algunos ejemplos que se pueden dar sobre el avance investigativo desarrollado en la universidad son los proyectos de titulación de diseño y construcción de un robot esférico [Chimarro, 2015], diseño y construcción de una plataforma robótica móvil para interiores [Gaona & Gallardo, 2014] y el diseño y construcción de un brazo robótico antropomórfico de siete grados de libertad con análisis cinemático y dinámico mediante algoritmos genéticos [Sánchez & Terán, 2015].

El proyecto de investigación es una oportunidad de desarrollar tecnología capaz de resolver problemas de manufactura y visión artificial mediante el uso de software y hardware libre. El estudio realiza un aporte importante para ampliar el horizonte de la robótica y sus alcances dentro de nuestra cultura aplicando los conocimientos impartidos con el diseño y construcción de un robot dual.

1.2. Justificación e Importancia

El desarrollo e implementación de robots en las industrias, se ha adaptado a determinados procesos y tareas consideradas peligrosas y con dificultad de acceso para el ser humano, o que se realizan en ambientes hostiles para el mismo. La implementación de estos robots requiere el cumplimiento de normas estrictas de seguridad en el trabajo que realizan, de esta manera se garantiza que el robot industrial puede realizar con total seguridad tareas iterativas, así como manejar productos potencialmente peligrosos o manipular cargas pesadas.

La influencia de la tecnología en el desempeño de la vida actual es de gran importancia en varios niveles del desempeño de la sociedad competitiva que se ha visto en la necesidad de desarrollar dispositivos cada vez más avanzados para dar solución a diferentes fenómenos y problemas ingenieriles a nivel industrial y social. El presente proyecto tiene como objetivo dar solución a problemas de origen ingenieril a nivel industrial en cuanto a inclusión de manipuladores robóticos como recurso factible frente a procesos repetitivos que pueden ser automatizados. Se propone también la optimización de tiempos de producción y liberación a operarios de actividades de alto impacto de carga y procedimientos iterativos o que necesiten medidas cerradas de precisión.

Siendo una de las áreas comprometidas al desarrollo de investigación actual la robótica, se propone el diseño e implementación de un manipulador robótico dual. El manipulador consta de un torso de dos brazos de siete grados de libertad cada uno. Mediante el proyecto de investigación se pretende incursionar en el estudio de tecnología

redundante y metodologías de diseño robótico, de esta forma se pretende la inclusión de los estudiantes al estudio de esta rama dejando a disposición de los estudiantes del Departamento de Ciencias de la Energía y Mecánica DECEM el dispositivo físico y toda la información que este pueda aportar para su formación académica.

1.3. Alcance

El proyecto a desarrollarse tiene como objetivo realizar un proceso automático de junta empernada mediante el uso de software libre y visión artificial. El manipulador constará de dos brazos antropomorfos de siete grados de libertad cada uno. El efector final será de tipo pinza acoplado una cámara para detección de imágenes. La estructura del robot constará de un torso y una base que serán el cuerpo y soporte del sistema. El desarrollo del proceso automático estará compuesto por un planificador encargado de realizar el control de cada uno de los brazos y supervisar el proceso. Se utilizará visión artificial como medio de retroalimentación para la detección de los elementos requeridos para desarrollar el proceso de junta empernada. El desarrollo de la secuencia se realiza mediante máquina de estados y posicionamiento offline.

1.4. Objetivos

1.4.1. Objetivo General

Diseñar y construir un torso robótico colaborativo de dos brazos, para la unión de una junta empernada en una celda de trabajo.

1.4.2. Objetivos Específicos

- Realizar el diseño Mecatrónico de un torso robótico, brazos y celda de trabajo que desarrollará un proceso de manufactura de junta empernada.
- Diseñar e implementar una placa PCB que contenga el circuito de potencia, circuito de sensores y actuadores y circuito de acondicionamiento de señales.
- Desarrollar e implementar una unidad de control jerárquica llamada Planificador que se encargará de administrar y controlar la posición de los dos brazos.
- Codificar la adquisición y tratamiento de imágenes del sistema de visión artificial ubicado en los efectores finales de cada brazo, para ejecutar el proceso de junta empernada.

1.5. Estructura del Documento

En el capítulo 1 se desarrolla los temas referentes a la descripción general del proyecto, los antecedentes desde un punto de vista global, el desarrollo de la tecnología a nivel mundial y su influencia a nivel micro. De acuerdo a la identificación del problema se plantea la justificación, el alcance del estudio en cada una de sus ramas y se delimita los objetivos del mismo.

En el capítulo 2 se realiza el estudio del arte acerca de la robótica industrial, su clasificación y aplicación de los mismos en función de su efector final. Además, se hace referencia a la importancia del estudio de la cinemática para el movimiento del robot y las técnicas de control disponibles para comandar su funcionamiento.

En el capítulo 3 se despliega la metodología y diseño mecatrónico del manipulador robótico colaborativo por medio de una estructura modular, cuyos módulos son: Estructura, Cinemática del manipulador, Visión artificial, y Planificador.

En el capítulo 4 se describe el procedimiento desarrollado para la implementación de cada uno de los módulos del sistema mecatrónico, y la inclusión de cada rama en un solo sistema. Posteriormente se presentan los resultados obtenidos de las pruebas realizadas en cada una de las fases de funcionamiento.

Capítulo 2

Estado del Arte

El capítulo presenta un resumen del concepto y clasificación de manipuladores robóticos industriales y sus campos de aplicación. Realiza una introducción a los robots colaborativos, técnicas de control de movimiento y optimización de trayectoria. Se menciona también la inclusión de un robot colaborativo dentro de una celda de trabajo y se hace referencia al estudio previo de un manipulador robótico antropomorfo de siete grados de libertad controlado mediante técnicas de algoritmos genéticos.

2.1. Manipuladores Robóticos Industriales

Un robot industrial es un manipulador multifuncional reprogramable, capaz de mover materias, piezas, herramientas, o dispositivos especiales, según trayectorias variables, programadas para realizar tareas diversas.

Existen cinco diferentes configuraciones de manipuladores dependiendo de la forma y movimiento de sus articulaciones, estas son:

- Cartesiano
- Cilíndrico
- Esférico
- Antropomorfo
- SCARA

2.1.1. Configuración Cartesiana

Posee tres movimientos lineales, es decir, tiene tres grados de libertad, los cuales corresponden a los movimientos localizados en los ejes X, Y y Z como se muestra en

la Figura 2.

Los movimientos que realiza este robot entre un punto y otro son con base en interpolaciones lineales. Interpolación, en este caso, significa el tipo de trayectoria que realiza el manipulador cuando se desplaza entre un punto y otro [Cepyme, 2016].

A la trayectoria realizada en línea recta se le conoce como interpolación lineal y a la trayectoria hecha de acuerdo con el tipo de movimientos que tienen sus articulaciones se le llama interpolación por articulación.

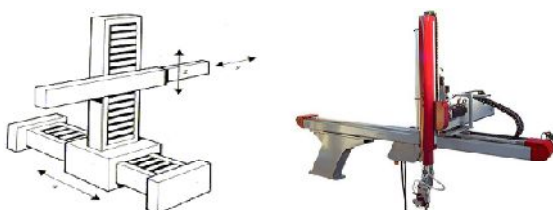


Figura 2: Robot Cartesiano
Fuente: [Cepyme, 2016]

2.1.2. Configuración Cilíndrica

Puede realizar dos movimientos lineales y uno rotacional, o sea, que presenta tres grados de libertad como se muestra en la Figura 3.

El robot de configuración cilíndrica está diseñado para ejecutar los movimientos conocidos como interpolación lineal e interpolación por articulación.

La interpolación por articulación se lleva a cabo por medio de la primera articulación, ya que ésta puede realizar un movimiento rotacional.

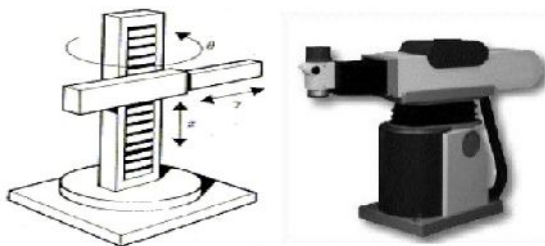


Figura 3: Configuración Cilíndrica.
Fuente: [Cepyme, 2016]

2.1.3. Configuración Polar

Tiene varias articulaciones. Cada una de ellas puede realizar un movimiento distinto: rotacional, angular y lineal.

Este robot utiliza la interpolación por articulación para moverse en sus dos primeras articulaciones y la interpolación lineal para la extensión y retracción como se muestra en la Figura 4.

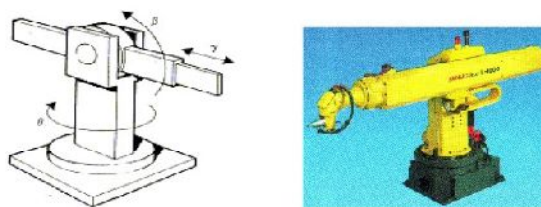


Figura 4: Configuración Polar.
Fuente: [Cepyme, 2016]

2.1.4. Configuración Angular

Presenta una articulación con movimiento rotacional y dos angulares. Aunque el brazo articulado puede realizar el movimiento llamado interpolación lineal (para lo cual requiere mover simultáneamente dos o tres de sus articulaciones), el movimiento natural es el de interpolación por articulación, tanto rotacional como angular como se muestra en la Figura 5.

Además de las cuatro configuraciones clásicas mencionadas, existen otras configuraciones llamadas no clásicas.

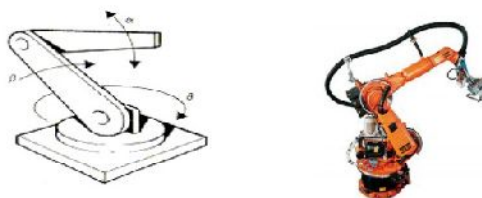


Figura 5: Configuración Angular.
Fuente: [Cepyme, 2016]

2.1.5. Configuración SCARA

El ejemplo más común de una configuración no clásica lo representa el robot tipo SCARA, cuyas siglas significan: Selective Apliance Arm Robot for Assembly. Este brazo puede realizar movimientos horizontales de mayor alcance debido a sus dos articulaciones rotacionales. El robot de configuración SCARA también puede hacer un movimiento lineal (mediante su tercera articulación) como se muestra en Figura 6.



Figura 6: Configuración SCARA.
Fuente: [Cepyme, 2016]

En la Tabla 1 se observa un resumen de las diferentes configuraciones, su esquema cinemático y espacio de trabajo.

Tabla 1.
Tipos de robots industriales según ISO8373

Configuración	Principio	Esquema cinemático	Volumen de trabajo	Foto
Cartesiano				
Cilíndrico				
Esférico				
SCARA				
Articulado				
Paralelo				

Fuente: [Gayo, 2013]

2.2. Manipuladores Colaborativos

A nivel industrial, según Montes describe en su artículo [Montes, 2015], en la actualidad el propósito de crear robots es incluirlos en el desarrollo de diferentes procesos y ponerlos al lado del trabajador. Dependiendo de las especificaciones necesarias para desarrollar cierta labor, el robot se acopla al ambiente de trabajo e interactúa con

los operarios. Además, las características mecánicas y componentes de recubrimiento también se adaptan a las necesidades del lugar de desempeño del manipulador.

Montes hace referencia a la presentación el robot colaborativo de dos brazos pensado para ser incluido en la industria electrónica. YuMi es uno de los ejemplos de implementaciones robóticas a nivel industrial existentes en la actualidad. Cuenta con catorce grados de libertad y una precisión de 0.02 milímetros, capaz de sujetar objetos sumamente frágiles, como se muestra en la Figura 7.



Figura 7: YuMi - robot colaborativo de dos brazos
Fuente: [ABB, 2015]

El robot manipulador necesita disponer de dos brazos para tener la función de colaboración al momento de ensamblar partes mecánicas, esta característica ha motivado a varias compañías robóticas a mejorar sus sistemas e incluir en ellos dos brazos y un torso, como se muestra en la Figura 8 Según el mencionado estudio, un sistema robótico de dos brazos tiene más ventajas que un manipulador tradicional.



Figura 8: Robot dual colaborativo de dos brazos
Fuente: [Park et al., 2010]

Park menciona que para realizar el diseño de movimiento, definir el alcance y volumen de revolución del manipulador colaborativo es necesario definir temas importantes para el desarrollo del trabajo del manipulador. En su trabajo de investigación analiza algunas técnicas de control de movimiento en función de las especificaciones del manipulador. Afirma que el nivel de complejidad de los movimientos del brazo robótico depende del orden de accionamiento de cada brazo. Propone en su estudio una técnica de enseñanza de trayectoria.

2.3. Selección de Manipuladores y Aplicación en la Industria

Existen numerosas aplicaciones dentro de la industria, donde es necesario el uso de máquinas para no poner en peligro al personal, se encargan de tareas pesadas y repetitivas:

- Ensamblado
- Soldadura
- Carga y descarga
- Corte y pulido
- Paletizado y empaclado
- Pintura

El manipulador designado a desarrollar determinada aplicación depende de las especificaciones y limitaciones existentes en el proceso. Algunas de las aplicaciones industriales en las que comúnmente se utiliza la ayuda robótica se pueden observar en la Figura 9.

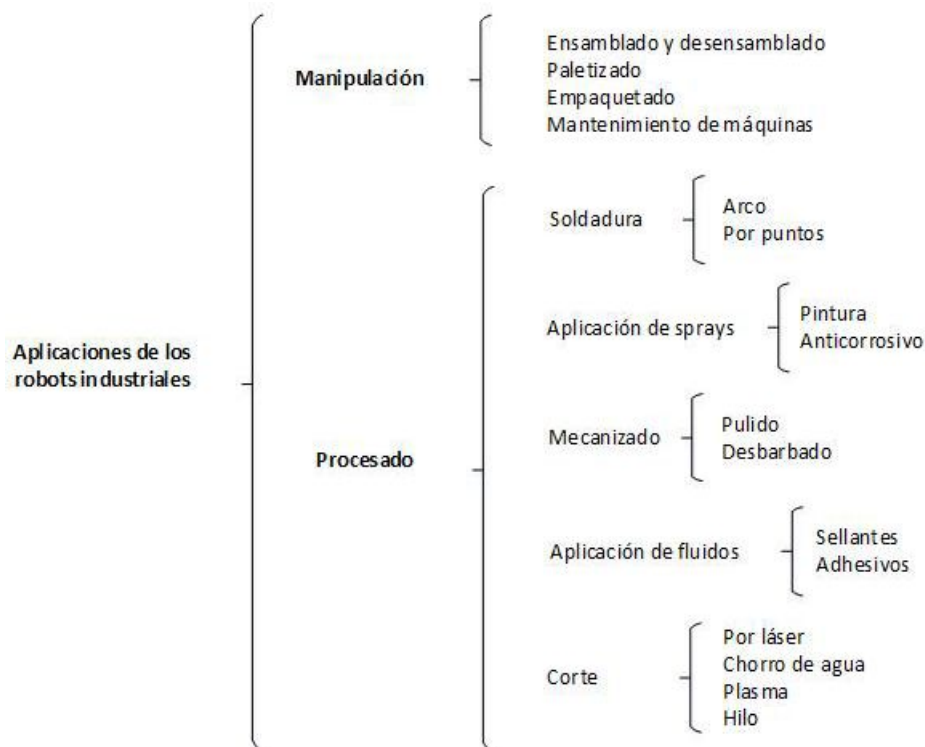


Figura 9: Diagrama de aplicaciones comunes de MRI en la industria.

Algunos ejemplos de implementación de manipuladores robóticos en la industria se pueden observar en las Figura 10.



Figura 10: Aplicaciones para a) Robot FANUC soldador, b) Robot KUKA en un sistema de paletización

Fuente: [Mecalux, 2015] [Rivas, 2010]

2.4. Efectores Finales

Un efector final es el dispositivo que se une a la muñeca del brazo del manipulador y permite darle al robot una tarea específica. Los efectores finales están directamente asociados a la aplicación que realiza el manipulador robótico. De acuerdo a la fuerza que ejerce sobre los componentes manipulados se pueden clasificar como se muestra en Figura 11:

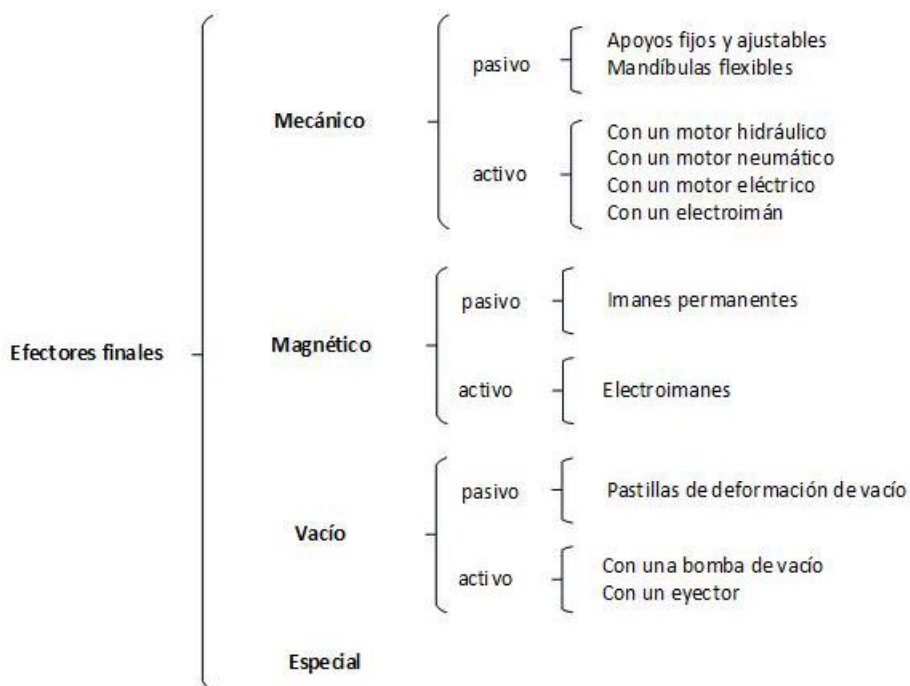


Figura 11: Clasificación de efectores finales.

Fuente: [Blecha et al., 2008]

Existen diferentes tipos de agarre de acuerdo a la forma del elemento a manipular y la aplicación que se necesita. La configuración del agarre es análoga a las diferentes formas de manipulación de objetos con una mano humana. En la Figura 12 se observa la clasificación de los tipos de agarre existentes para un efector final.

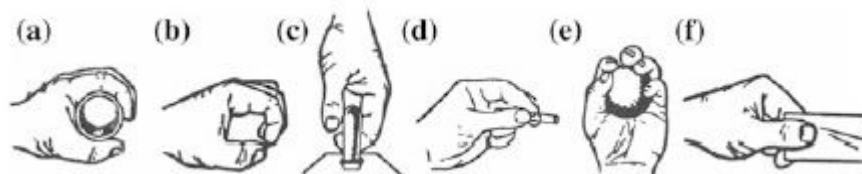


Figura 12: Tipos de agarre humano a) cilíndrico b) entre los extremos c) gancho d) palmar e) esférico f) lateral.

Fuente: [Ceccareli, 2013]

Para realizar la manipulación de un objeto con dos dedos es posible considerar diferentes configuraciones. La configuración a elegir está relacionada con la posición relativa de los dedos y la fuerza de intervención. Los tipos de configuraciones de este tipo se muestran en la Figura 13 y pueden ser:

- Planar: cuando todas las fuerzas de accionamiento están situadas en el mismo plano.
- Espacial: cuando las fuerzas se encuentran en diferentes planos.
- Palmar: cuando el peso del objeto se dirige a la palma de la pinza o su superficie.
- Con carga en un dedo: cuando el peso del objeto está en un dedo.

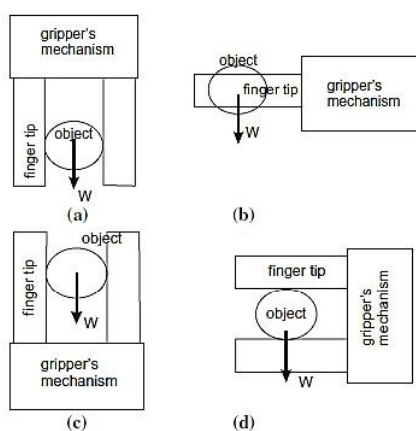


Figura 13: Configuraciones de agarre a) planar b) espacial c) palmar d) agarre con carga.

Fuente: [Ceccareli, 2013]

2.5. Celda de Trabajo

Para identificar la distribución de la zona de trabajo de un manipulador colaborativo de dos brazos es importante conocer varias características. Se realiza un análisis del área que ocupa el manipulador en la posición home definida, el alcance máximo dimensional de sus brazos y su volumen de trabajo, el proceso que desarrolla y las necesidades espaciales para el correcto desempeño de cada uno de sus movimientos. En el estudio desarrollado por Park, se realiza el análisis del diseño del manipulador para identificar su zona de trabajo y definir el espacio cooperativo disponible. El volumen de trabajo y la disposición de las diferentes áreas de movimiento de cada brazo independiente se muestran en la Figura 14

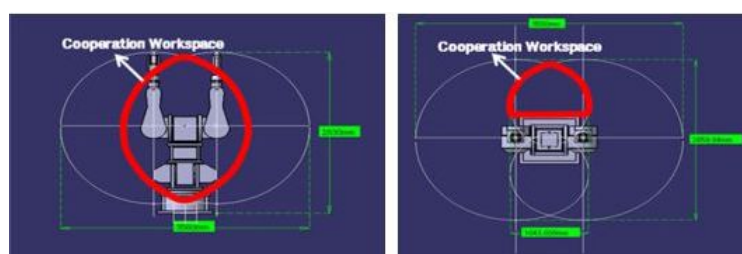


Figura 14: Zona de trabajo cooperativo del manipulador
Fuente: [Park et al., 2010]

La disposición de la celda de trabajo del manipulador colaborativo de dos brazos se ve influenciada por varios aspectos. El análisis dimensional del robot, sus características físicas, alcances máximos y mínimos, el análisis de diseño y cinemático forman parte importante de la disposición del lugar de trabajo. En la Figura 15 se observa un ejemplo del layout de un robot colaborativo de dos brazos en función de sus características mecánicas y el trabajo de manufactura que desarrolla.

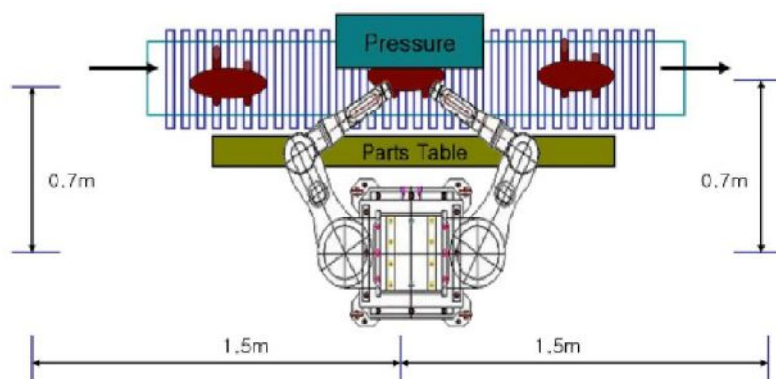


Figura 15: Layout de un robot colaborativo de dos brazos
Fuente: [Park et al., 2010]

2.6. Cinemática de Manipuladores Robóticos

En el artículo cinemática de Manipuladores [Jaramillo, 2005] describe a la cinemática como el estudio de los movimientos de un robot en función del tiempo. El estudio de la cinemática involucra por lo tanto las posiciones, velocidades y aceleraciones, sin considerar las fuerzas que originan dicho movimiento. El problema cinemático se divide en dos, como se muestra en la Figura 16:

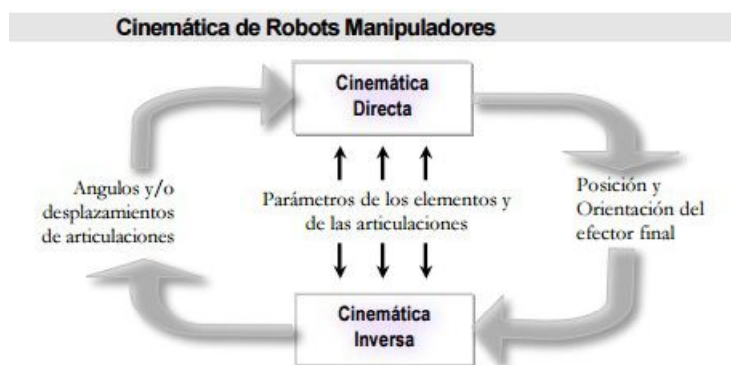


Figura 16: Problema cinemático.

Fuente: [Jaramillo, 2005]

2.6.1. Cinemática Directa

A partir de los parámetros físicos, que definen la geometría de un manipulador dado, y de los ángulos en el caso de articulaciones de tipo rotacional y/o desplazamientos articulares en el caso de articulaciones de tipo traslacional, se halla la posición y orientación del efector final en el espacio tridimensional, como se muestra en la Figura 17.

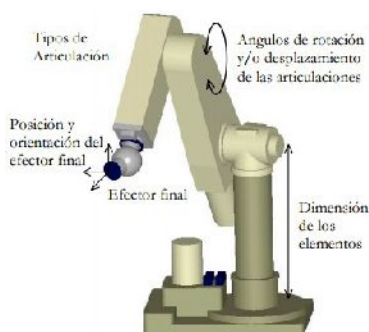


Figura 17: Problema cinemático en manipuladores, representación PUMA 260

Fuente: [Jaramillo, 2005]

2.6.1.1. Aspectos Importantes en la Solución de un Problema Cinemático

[Jaramillo, 2005] describe los siguientes aspectos:

- El desplazamiento espacial total del efector final se debe las rotaciones/traslaciones de los elementos anteriores (dado un sistema de coordenadas de referencia). Esto se asume debido a que un manipulador es un conjunto de cuerpos ligados o acoplados por articulaciones.
- Denavit y Hartenberg (1955) proponen un método para describir y representar la geometría espacial de los elementos de un brazo con respecto a un sistema de coordenadas fijo a través de matrices de transformación de 4×4 , y reducen el problema directo a encontrar la relación entre el desplazamiento espacial del sistema de coordenadas del efector final y el sistema de coordenadas de referencia.

2.6.1.2. Resolución del Algoritmo de Denavit Hartenberg

Para la resolución del algoritmo de Denavit Hartenberg es necesario seleccionar un sistema de referencia a los eslabones y posteriormente se procede a realizar el algoritmo como tal.

Selección del sistema de referencia

Inicialmente se describe la convención para asignar los sistemas de referencia cartesianos asociados a los eslabones del robot. En la literatura de la robótica, la convención Denavit-Hartenberg no es única, depende de la selección de los sistemas de referencia cartesianos en las articulaciones y eslabones, así como en sus eslabones adyacentes.

La cinemática directa del robot proporciona las coordenadas cartesianas del extremo final del robot relativo a un sistema de referencia cartesiano fijo $\sum_0(x_0, y_0, z_0)$.

En general se tiene el procedimiento dado por:

- El eje z_i se asigna rígidamente a la articulación $i + 1$. Es decir, z_0 es el eje de la articulación 1, z_1 es el eje de la articulación 2, y así sucesivamente.
- Localizar el origen o_i del sistema de referencia $\sum_i(x_i, y_i, z_i)$ en la intersección del eje z_i con la normal común a los ejes z_{i-1} y z_i .
- Seleccionar el eje x_{i-1} sobre la normal que une los ejes z_{i-1} y z_i en dirección de la articulación $i - 1$ hacia la articulación i .
- Definir el ángulo de torsión α_i , este es el ángulo entre los ejes z_i y z_{i-1} y se mide con valor positivo en el sentido de las manecillas del reloj sobre el eje x_i .

- Seleccionar el eje y_i por la regla de la mano derecha.

En la Figura 18 se observa la distribución de los ejes correspondientes a la cadena cinemática según la convención definida por el algoritmo.

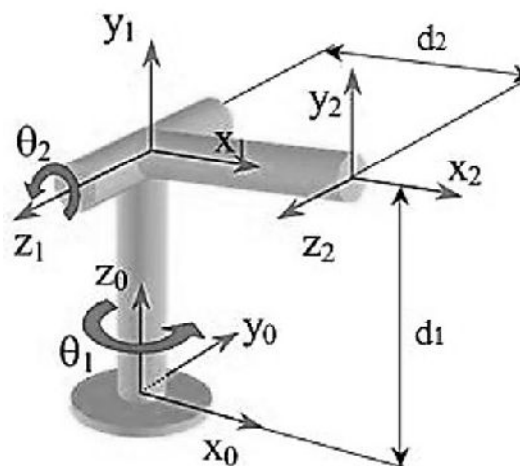


Figura 18: Ejes convencionales para uso de algoritmo de Denavit Hartenberg

La convención Denavit Hartenberg proporciona una representación no única para los siguientes casos:

- Para el sistema de referencia $\sum_0(x_0, y_0, z_0)$ sólo la dirección del eje z_0 es especificada, entonces su origen o_0 y el eje x_0 pueden ser seleccionados de manera arbitraria.
- Para el sistema de referencia $\sum_n(x_n, y_n, z_n)$ la articulación $n + 1$, entonces el eje z_n no está completamente definido, mientras que el eje x_n es normal al eje z_{n-1} . Típicamente la n -ésima articulación es rotatoria, por lo tanto z_n se alinea en la dirección de z_{n-1} .
- Cuando dos ejes consecutivos z_i y z_{i-1} son paralelos entre sí, la normal común entre ellos no es única.
- Cuando dos ejes consecutivo z_i y z_{i-1} se interceptan, la dirección del eje x_i es arbitraria.
- Cuando la articulación i -ésima es lineal o prismática, entonces la dirección de z_i es arbitraria.

Pasos para resolver el algoritmo de Denavit Hartenberg

Localizar la dirección de los ejes z_0, z_1, \dots, z_{n-1} .

Establecer el sistema de referencia cartesiano fijo $\Sigma_0(x_0, y_0, z_0)$ cuyo origen es colocado sobre el sistema de referencia en la base del robot. Los ejes x_0, y_0 son determinados de acuerdo con la regla de la mano derecha. Una vez que el sistema de referencia $\Sigma_0(x_0, y_0, z_0)$ ha sido establecido, se inicia un proceso iterativo en el cual se define el sistema de referencia $\Sigma_i(x_i, y_i, z_i)$ usando el sistema de referencia $\Sigma_{i-1}(x_{i-1}, y_{i-1}, z_{i-1})$, iniciando con el sistema de referencia $\Sigma_1(x_1, y_1, z_1)$.

Localizar el origen o_i en la intersección de la normal común que une al eje z_i con el eje z_{i-1} . Si el eje z_i intercepta al eje z_{i-1} colocar o_i en la intersección. Para el caso en que los ejes z_i y z_{i-1} son paralelos: Si la articulación i -ésima es rotacional, colocar el origen o_i sobre la articulación i -ésima, tal que $d_i=0$. Si la articulación i -ésima es prismática, colocar el origen o_i en un punto límite físico de la articulación i -ésima, por ejemplo en un punto extremo.

Seleccionar el eje x_i a lo largo de la normal común que une a los ejes z_{i-1} y z_i , en dirección de la articulación $i-1$ hacia la articulación i .

Determinar y_i por la regla de la mano derecha.

Establecer el sistema de referencia del extremo final $\Sigma_n(x_n, y_n, z_n)$.

Si la articulación n -ésima es rotatoria, entonces alinear el eje z_n con el eje z_{n-1} . Si la articulación n -ésima es prismática, entonces seleccionar el eje z_n de forma arbitraria. El eje x_n debe cumplir el paso 4.

2.6.2. Cinemática Inversa

A partir de un conjunto de parámetros físicos, que definen la geometría de un manipulador dado, y de una posición y orientación específica del efector final, se halla un conjunto de ángulos y/o desplazamientos articulares que dan como resultado dicha posición y orientación del efector final.

[Ramírez, 2012] define la cinemática inversa como la solución de la configuración que debe adoptar el robot para llegar a una posición y orientación del extremo conocidas. Es decir que dada una posición del efector final y la longitud de cada eslabón, se debe encontrar los ángulos de las articulaciones. El procedimiento de obtención de las ecuaciones es dependiente de la configuración del robot. Para obtener la cinemática inversa es necesario resolver un conjunto de ecuaciones algebraicas no lineales simultáneas. La cinemática inversa presenta problemas en la resolución como:

- Ecuaciones no lineales.
- La existencia de múltiples soluciones.
- La posibilidad de que no exista una solución.

- Singularidades.

No existen algoritmos generales de solución al problema de cinemática inversa, pero [Ramírez, 2012] describe algunos tipos de solución que se utilizan generalmente:

2.6.2.1. Métodos Geométricos

Es un procedimiento adecuado para robots de pocos grados de libertad, o para el caso de que solo se consideren solo los primeros grados de libertad, dedicados a posicionar el extremo. Se suele utilizar para las primeras variables articulares, uso de relaciones geométricas y trigonométricas.

El método se basa en encontrar un número suficiente de relaciones geométricas en las que intervendrán las coordenadas del extremo del robot, sus coordenadas articulares y las dimensiones físicas de sus elementos.

2.6.2.2. Desacoplamiento Cinemático

En general no basta con posicionar el extremo del robot en un punto del espacio, sino que es preciso conseguir que la herramienta se oriente de una manera determinada. Para ello, los robots cuentan con otros tres grados de libertad adicionales, situados al final de la cadena cinemática y cuyos ejes, generalmente, se cortan en un punto, que informalmente se denomina muñeca del robot. Si bien la variación de estos tres últimos grados de libertad origina un cambio en la posición final del extremo real del robot, su verdadero objetivo es poder orientar la herramienta del robot libremente en el espacio.

El método de desacoplo cinemático separa posición y orientación. Para ello, dada una posición y orientación final deseadas, establece las coordenadas del punto de corte de los 3 últimos ejes (muñeca del robot) calculándose los valores de las tres primeras variables articulares (q_1 , q_2 , q_3) que consiguen posicionar este punto. A partir de los datos de orientación y de los ya calculados (q_1 , q_2 , q_3) se obtiene los valores del resto de las variables articulares.

2.6.2.3. Algoritmos Genéticos

En la investigación realizada por [González, 2004], se describe a los Algoritmos Genéticos AG están basados en la mecánica de la selección y la genética natural, las cuales rigen el comportamiento de los seres vivos. Las investigaciones realizadas por Charles Darwin demuestran que las especies más prominentes de evolución, cambian para adaptarse a su ambiente, sobrevivir y prosperar. Las especies no cambian por sí mismos, pero aquellos que mejor sobreviven, o tienen aptitud más alta, sobrevivirán más tiempo y producirán más descendencia. Es esta mejora continua la que inspira

a los científicos de los computadores a crear los AG. Un AG, se dice que está en la capacidad de imitar este proceso si está adecuadamente codificado.

Además [González, 2004], recalca que en la naturaleza aquellas especies que tengan más éxito en cuanto a sobrevivir y reproducirse tendrán un número mayor de descendientes; y por el contrario los que tengan bajo desempeño, tendrán pocos o ningún descendiente. Lo anterior descrito, visto desde el punto genético significa que los genes de los individuos más adaptados se propagarán hacia un número creciente de individuos en cada generación sucesiva. De esta manera las especies evolucionan para llegar a estar más adaptadas a su ambiente.

Los AG usan una analogía directa del comportamiento natural. Trabajan con una población de “individuos”, donde cada uno representa una solución posible a determinado problema. A cada individuo se le asigna un puntaje asociado a su “adaptación”, según sea su proximidad a la solución del problema. A estos “individuos”, se les da entonces la oportunidad de reproducirse con otros miembros de la población y sus descendientes comparten la información genética de sus dos padres. Los miembros menos adaptados de la población tienen por consiguiente menos probabilidad de reproducirse y entonces morirán más rápidamente. De lo anterior se puede entonces concluir que los AG no son más que una metáfora de un proceso natural.

Los AG pertenecen a la Inteligencia Artificial IA, en esta área los AG aparecen como una particularidad de lo que actualmente se conoce como Computación Evolutiva CE. Dentro de las diversas tendencias de la CE, la programación evolutiva nace con D. Fogel en 1960, quién define que el objetivo de esta disciplina es lograr un comportamiento inteligente a partir de la simulación de la evolución. La propuesta es operar sobre máquinas de estado finitas y las correspondientes representaciones discretas.

La investigación de [González, 2004], explica que los genes codifican un autómata finito o máquina de estados. Cada gen puede consistir en una terna del tipo “estado inicial + cadena de transiciones + estado final”. Es decir, cada gen es una regla que especifica las condiciones, llamadas cadena de transiciones, que se deben cumplir para llegar a un estado final. La cadena de transiciones corresponde con los sentidos o sensores del autómata el suceso, y los estados corresponden con las acciones o actuadores. Esta disciplina enfatiza los nexos de comportamiento entre padres e hijos, en lugar de buscar emular operadores genéticos específicos.

El algoritmo básico de la programación evolutiva es el siguiente:

1. Generar aleatoriamente una población inicial.
2. Aplicar el operador mutación.
3. Calcular la aptitud de cada hijo y usar un proceso de selección mediante el torneo

(normalmente estocástico) para determinar cuáles serán las soluciones que se retendrán.

4. Volver al segundo paso, si no se ha cumplido el criterio de parada.

El algoritmo básico para los AG es el siguiente:

1. Generar aleatoriamente una población inicial.
2. Calcular aptitud de cada individuo.
3. Seleccionar probabilísticamente con base en la aptitud individual.
4. Aplicar operadores genéticos (cruce y mutación), para generar la siguiente población.
5. Evaluar condición de parada, si no cumple regresar al segundo paso.

Algunas aplicaciones de los AG son las siguientes:

- Aprendizaje de máquina (sistemas clasificadores).
- Reconocimiento de patrones (por ejemplo, imágenes).
- Predicción.
- Planeación de movimientos de robots.

2.7. Generación y Optimización de Trayectoria para Robots Colaborativos

En el artículo de planificación óptima de trayectoria de [Hao et al., 2004], se propone un método óptimo de planificación de trayectorias con reducción de vibraciones para un robot espacial de brazo dual con enlaces flexibles delanteros usando un enjambre de partículas de optimización (PSO). la trayectoria de movimiento se describe mediante el uso de un spline de cuarto orden, con puntos de control como parámetros a ser optimizados.

[Hao et al., 2004] describe además a varios autores y sus formas de optimizar la trayectoria de robots colaborativos duales. Zhao y Chen eligen los sinusoides como funciones de base y la suma ponderada de ellos como la trayectoria de un robot. Kojima Kibe usa funciones polinómicas cúbicas para describir los conjuntos de velocidades angulares, y el índice de rendimiento se construye mediante desplazamientos de

flexión en los puntos finales de enlaces flexibles. Al optimizar el índice de rendimiento utilizando un algoritmo genético, se obtienen los parámetros óptimos de funciones polinómicas. Dado que los índices de rendimiento en ambos métodos se obtienen a través de la simulación de un proceso de movimiento del robot, las trayectorias resultantes están sujetos a los métodos de control utilizados.

2.8. Técnicas de Control para Robots Colaborativos

Existen varias técnicas de control para determinar el desempeño del movimiento del manipulador. Entre las metodologías básicas existen:

- Control punto a punto: consiste en determinar una serie de puntos en el espacio de trabajo del manipulador por donde se desea que el extremo final pase, con el fin de satisfacer cierta aplicación como se muestra en la Figura 19 .

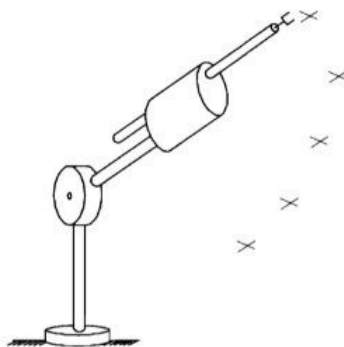


Figura 19: Control punto a punto.
Fuente: [Kelly & Santibáñez, 2003]

- Trayectoria continua: se determina una curva o trayectoria parametrizada temporalmente en el espacio de trabajo y hacer pasar el extremo del manipulador por dicha trayectoria tan cerca como sea posible, como se puede observar en la Figura 20.

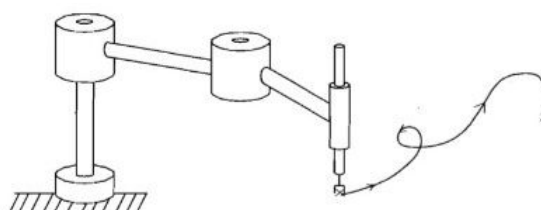


Figura 20: Control por trayectoria.
Fuente: [Kelly & Santibáñez, 2003]

El tema de control de movimiento conocido también como navegación de robot consiste en resolver tres subsistemas según [Kelly & Santibáñez, 2003]:

- Planeación del itinerario: consiste en determinar una curva en el espacio de trabajo del robot que conecte, sin tocar obstáculos, la posición inicial del extremo con la posición final deseada.
- Generación de trayectoria: consiste en parametrizar en el tiempo la curva obtenida en la planeación del itinerario.
- Diseño del controlador: trata de resolver el problema de control de movimiento previamente enunciado.

Entre las técnicas de control más utilizadas para resolver el problema de control de movimiento de manipuladores se tiene:

- Control adaptable
- Control con estructura variable
- Control por aprendizaje
- Control mediante lógica borrosa
- Control neuronal
- Control por máquina de estados

Cada una de estas técnicas de control es utilizada en los manipuladores de acuerdo a la aplicación que se desea en su desempeño. Dependiendo del proceso que realizará el manipulador, se elige la técnica de control más adecuada. Sin embargo, una de las técnicas eficientes para el desarrollo de aplicaciones que desarrollan procesos automáticos es el control por máquina de estados.

2.8.1. Planificador y Control Jerárquico

En el artículo de [Herrero et al., 2016] se describe al control jerárquico como programación de robots bi-brazo. La programación de robots bi-brazo se presenta en dos escenarios diferentes: trabajos no coordinados y trabajos coordinados.

Los trabajos no coordinados, cada brazo trabaja en un proceso diferente. Un brazo no tiene que preocuparse de los movimientos que realice el otro. En trabajos coordinados ambos brazos realizan diferentes tareas del mismo proceso, existen dos configuraciones: coordinados por objetivo y coordinación bi-brazo. Para estas configuraciones

es necesario la comunicación o la coordinación entre los brazos para poder asegurar el correcto comportamiento de todo el sistema.

En el caso de que estén coordinados por objetivo los brazos no interactúan entre ellos, pero tienen ciertos elementos en común, por ejemplo que estén paletizando en la misma caja. Para la coordinación bi-brazo, es necesario controlar y coordinar sincronamente el robot completo, por ejemplo, mover elementos manipulándolos con los dos brazos, o procesos en los que un brazo sujeta un elemento y el otro realiza una operación de ensamblaje, como en la Figura 21.



Figura 21: Robot Kawada Nextage Open de Tecnalía. Robot bi-brazo
Fuente: [Herrero et al., 2016]

2.9. Sensado y Sistema de Visión Artificial

En el artículo de visión artificial del [Ministerio de Educación, 2012], se describe al sistema de visión artificial como un campo de inteligencia artificial, mediante la utilización de técnicas adecuadas, permite la obtención, procesamiento, y análisis de cualquier tipo de información especial obtenida a través de imágenes digitales.

Las aplicaciones de Visión Artificial industrial integran sistemas de captura de imágenes digitales, dispositivos de entrada/salida y redes de ordenador para el control de equipos destinados a la fabricación, tales como brazos robóticos. Los sistemas de Visión Artificial se destinan a realizar inspecciones visuales que requieren alta velocidad, gran aumento, funcionamiento las 24 horas del día o la repetibilidad de las medidas. El objetivo de un sistema de inspección por Visión Artificial suele ser comprobar la conformidad de una pieza con ciertos requisitos, tales como las dimensiones, números de serie, la presencia de componentes, etc.

La adquisición de imágenes digitales requiere de dos elementos básicos:

1. Dispositivo físico sensible a una determinada banda del espectro de energía electromagnético que genera una señal eléctrica.

2. Digitalizador: dispositivo para convertir la señal eléctrica continua en un conjunto discreto de localizaciones en el plano de la imagen.

La digitalización de imágenes se realiza mediante un proceso que inicia desde la escena hasta el producto final que es la imagen digitalizada y lista para procesar como se observa en la Figura 22.



Figura 22: Digitalización de imágenes.

Existen dos tipos de tecnología para la captura y digitalización de imágenes:

- CCD: La tecnología Charge Coupling Device es la que proporciona una mejor imagen, sobre todo en el caso de las cámaras profesionales en las que se emplea un sensor para cada color, mientras que en las de consumo se emplea un único sensor, de modo que este está constituido como un mosaico de detectores de colores y por tanto captura una menor densidad de puntos. Cuando el sensor ha capturado la imagen envía la información completa serializada fila a fila.
- CMOS: es la tecnología más económica, y ofrece un menor tamaño, gracias a su grado de integración electrónica. Además, puede capturar partes de la imagen sin tener que transmitirla completamente. Al contrario que en los sensores de tecnología CCD, las celdas de la matriz CMOS son totalmente independientes de sus vecinas. La principal diferencia radica en que en estos sensores la digitalización se realiza píxel a píxel dentro del mismo sensor, por lo que la circuitería del sensor es mucho más sencilla.

Debido a las características que proporcionan estas dos tecnologías, el uso que se le da a cada una de ellas es para diferentes aplicaciones. Dependiendo de la calidad y fidelidad de la imagen necesitada, se utiliza cada tecnología para ejemplos como los que se observan en la Figura 23.



Figura 23: Uso de tecnologías CCD y CMOS.

El procesamiento y análisis de imágenes se compone de varios pasos, siendo estos los que se muestran en la Figura 24. La imagen parte de la captura de la escena de interés, continua con su acondicionamiento. Se separan y extraen los componentes principales para poder diferenciar la diversidad de objetos y formas con sus dimensiones características.

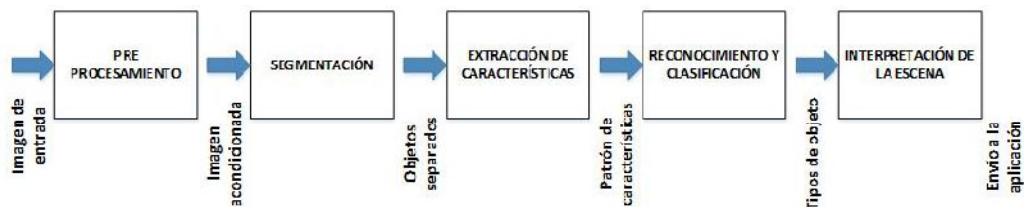


Figura 24: Procesamiento y análisis de imágenes

Las diferentes áreas desarrolladas por un sistema de visión artificial se pueden definir según [Porrás & De la Cruz, 2015] como:

- Captación: Proceso de obtener una imagen. El proceso de captura está centrado en tres aspectos importantes, necesarios para obtener una imagen de calidad que no altere las condiciones reales agregando errores considerables al sistema.
- Pre procesamiento: Métodos para quitar o reducir características no deseadas en la imagen como el ruido. El pre-procesamiento consiste en la aplicación de técnicas que permitan el realce o mejoramiento de algunas características importantes en las imágenes originales para facilitar el proceso de segmentación.

- Segmentación: Dividir una imagen en objetos que sean de nuestro interés. La técnica de segmentación más conocida dentro del procesamiento de imágenes es quizás la umbralización. Este método permite separar dos o más regiones de una imagen a partir de un análisis del histograma.
- Descripción: Obtención de características de una objeto como su forma, tamaño, color con el fin de diferenciarlo de otros objetos.
- Reconocimiento: Proceso a través del cual se identifica un objeto dentro de una escena.
- Interpretación: Asociar un significado a un conjunto de objeto reconocidos.

2.9.1. Detección de Geometría

Se puede hacer uso de un sistema de visión artificial para diferentes aplicaciones, una de estas es el reconocimiento de formas. Como se trata en el estudio de Porras [Porras & De la Cruz, 2015], la detección de geometría de un objeto parte de la captura de la escena y posteriormente a la digitalización de la imagen como se muestra esquemáticamente en la Figura 25.

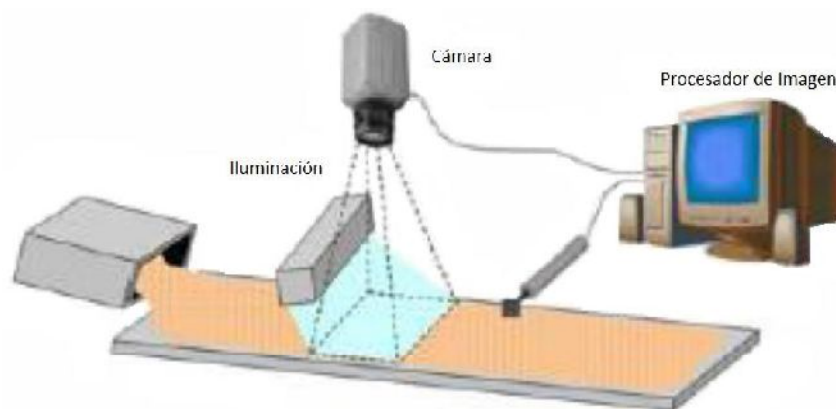


Figura 25: Detección de imágenes y proceso de digitalización
Fuente: [Porras & De la Cruz, 2015]

De acuerdo a la aplicación para la cual es requerida la imagen, se realiza el proceso de filtrado y digitalización de la imagen siguiendo una serie de pasos, los cuales pueden contener una combinación de filtros y detección de bordes. Un ejemplo claro del proceso de detección de formas se observa en la Figura 26.

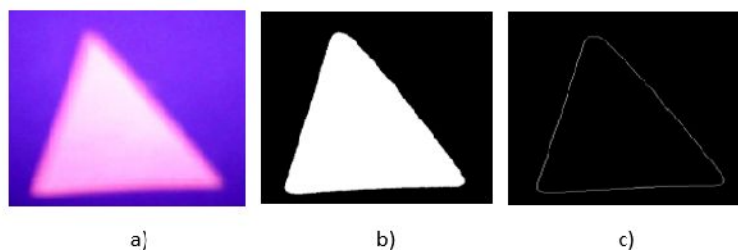


Figura 26: Detección de formas a) Imagen original b) Imagen convertida a binaria c) Imagen filtrada para detección de bordes

Fuente: [Porras & De la Cruz, 2015]

Mediante el uso de software libre, existen varias alternativas a las cuales se pueden recurrir en cuanto a la detección de objetos y reconocimiento de formas se trata. Una de las opciones es el uso de OpenCV [OpenCV, 2016]. Open CV (Open Source Computer Vision) es una librería de visión por computador de código abierto. Tiene interfaces de C ++ , C, Python y Java y es compatible con Windows , Linux , Mac OS , iOS y Android . OpenCV fue diseñado para la eficiencia computacional y con un fuerte enfoque en aplicaciones en tiempo real. La investigación desarrollada para el control de un cuadricóptero utilizando ROS [Calderón, 2014] hace uso de la librería OpenCV para el seguimiento de escenas y procesamiento morfológico. En el estudio desarrollado se identifican formas básicas como se muestra en la Figura 27.



Figura 27: Ejemplo de elementos estructurales típicos

Fuente: [Calderón, 2014]

El procesamiento morfológico se hace necesario para clasificar de forma correcta las diferentes formas y geometría de los objetos en estudio. Calderón propone un diagrama de flujo como se muestra en la Figura 28 de forma que se pueda realizar un correcto procesamiento de la imagen para identificación de geometría de objetos capturados.

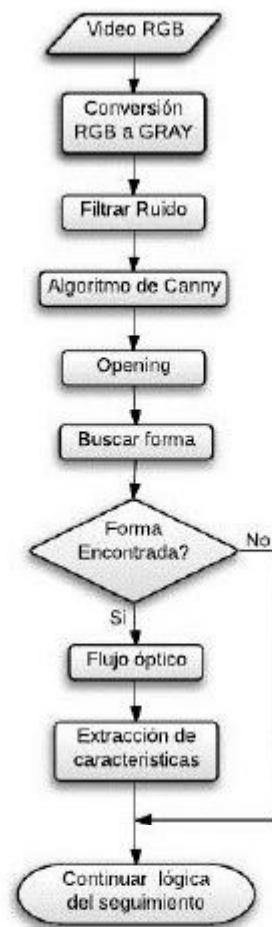


Figura 28: Diagrama de flujo para reconocimiento por forma
Fuente: [Calderón, 2014]

Calderón da a conocer que al usar las funciones de contornos que proporciona OpenCV se puede obtener una secuencia de puntos de vértices y además se identifican algunas figuras según el número de vértices. Además, propone la aproximación poligonal de imágenes haciendo uso de funciones proporcionadas por OpenCV además de las opciones de dilatar y erosionar imágenes.

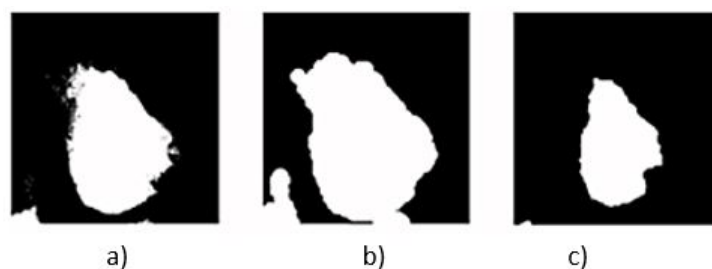


Figura 29: Imágenes a) original b) dilatada c) erosionada
Fuente: [Calderón, 2014]

2.10. Manipulador Antropomorfo de Siete Grados de Libertad

El proyecto desarrollado por [Sánchez & Terán, 2015], define al manipulador antropomórfico de siete grados de libertad como un robot redundante. Un robot redundante es aquel que posee más grados de libertad que los que usa convencionalmente para las tareas que pueda realizar el mismo como manipulación, soldadura, pintura, entre otras. Este tipo de configuración posee más de seis grados de libertad y son usados para lugares de difícil acceso, mayor flexibilidad y seguridad industrial. La configuración es más humanoide, permitiendo libertad de movimientos y flexibilidad, como se lo puede visualizar en la Figura 30.

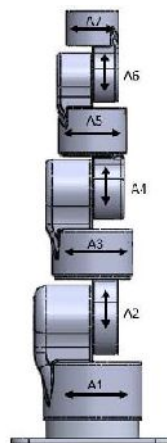


Figura 30: Configuración robot redundante.
Fuente: [Sánchez & Terán, 2015]

La configuración desarrollada por [Sánchez & Terán, 2015] es de siete eslabones y siete articulaciones rotacionales alternadas una alrededor del eje Z y la siguiente alrededor del eje X y así sucesivamente hasta terminar con la última articulación girando con respecto al eje Z como se puede ver en la Figura 31; debido a que permite una mayor flexibilidad de movimientos y es adecuada para demostrar el uso de posicionamiento en un robot redundante además posee una serie de aplicaciones industriales como son manipulación en espacios complicados, robots de asistencia, etc.



Figura 31: Vista Isométrica realizada esquemáticamente con RoboAnalyzer, del robot antropomórfico de 7 GDL.

Fuente: [Sánchez & Terán, 2015]

Debido al número de grados de libertad, es compleja la resolución de la cinemática y dinámica del robot. Es complicado resolver el sistema por medio de geometría o por medio del método de matriz homogénea. Actualmente se utilizan métodos evolutivos como algoritmos genéticos para resolver el problema cinemático inverso.

2.10.1. Características del Manipulador Antropomórfico de Siete Grados de Libertad

El robot antropomórfico desarrollado por [Sánchez & Terán, 2015] consta de:

- Siete grados de libertad.
- Sus ejes se distribuyen como se muestra en la Figura 30.
- Capacidad de carga pequeña, debido a que solo es para fines didácticos.
- Utiliza actuadores eléctricos.
- Alcance mediano de 40 a 60 cm.
- Realiza actividades similares a un brazo humano.
- Su efector final es una pinza que permite manipular objetos.

2.10.2. Análisis Cinemático y Dinámico mediante Algoritmos Genéticos

En la investigación de cinemática inversa de la Universidad Nacional de Colombia [Giraldo et al., 2006] se describe la resolución de cinemática inversa partiendo de la solución de cinemática directa de Denavit Hartenberg, luego se introduce un algoritmo genético básico, variando la función de aptitud. Los resultados obtenidos en el Scorbot

VRplus de cinco grados de libertad, demostró que el algoritmo genético es efectivo para resolver el problema y genera varias soluciones al encontrar la cinemática inversa.

El proyecto [Sánchez & Terán, 2015], asegura que el método de algoritmos genéticos gana importancia en el ámbito de la robótica y ha obtenido buenos resultados. En dicho proyecto se diseña y construye un robot de siete grados de libertad debido a sus amplias aplicaciones y flexibilidad, obteniendo la solución de su posicionamiento y orientación que le permiten realizar aplicaciones de manipulación flexibles.

El robot posee siete grados de libertad y por ello se presentan varias posibilidades de movimiento para alcanzar un punto. Pese a que el algoritmo genético se diseña para que realice el mínimo cambio articular, el robot puede presentar trayectoria con colisiones.

2.10.2.1. Parámetros de Denavit Hartenberg

Para realizar el análisis cinemático, es necesario obtener los parámetros Denavit Hartenberg que representan al robot desarrollado por [Sánchez & Terán, 2015] de la Figura 32 y cuyos resultados se muestran en la Tabla 2.

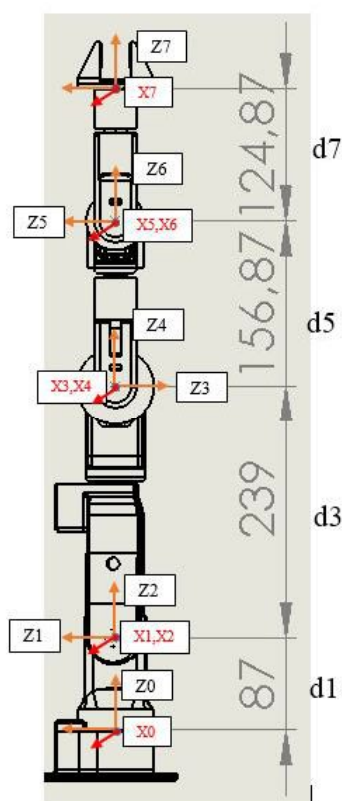


Figura 32: Diagrama que representa los parámetros de Denavit Hartenberg.

Fuente: [Sánchez & Terán, 2015]

Tabla 2.
Parámetros de Denavit Hartenberg del robot AR7D

Eslabón	a	α	D	ϑ
1	0	90°	8.7	Q1
2	0	-90°	0	Q2
3	0	-90°	23.9	Q3
4	0	90°	0	Q4
5	0	90°	15.687	Q5
6	0	-90°	0	Q6
7	0	0°	12.487	Q7

Fuente: [Sánchez & Terán, 2015]

2.10.2.2. Solución de Cinemática Inversa mediante Algoritmo Genético

[Sánchez & Terán, 2015] utilizó el toolbox de Matlab gatool el cual nos permite optimizar y variar los parámetros de algoritmo genético hasta que el resultado sea favorable. El algoritmo genético de Matlab trabaja de la siguiente manera:

1. Recibe parámetros iniciales como número de variables, restricciones y límites de las variables, los límites serían el máximo y mínimo ángulo que puede alcanzar el servomotor o el ángulo máximo que no produzca una colisión, esto permitirá que las variables estén siempre dentro de este rango y no existan ángulos inalcanzables para el robot.
2. Crea una población inicial que puede ser controlada o aleatoria, evalúa los individuos y selecciona los mejores.
3. Crea una secuencia de nuevas generaciones tomando en cuenta parámetros como la aptitud, elitismo, generando padres que se cruzan o mutando padres solteros, por último reemplazando la población por la nueva población o hijos.
4. Se detiene cuando llega a un criterio de parada que puede ser el máximo número de generaciones, haber alcanzado un error aceptable, o que ya no varíen los individuos a lo largo de varias generaciones.

Se presentan restricciones, una de las ventajas de los algoritmos genéticos es que se puede restringir el espacio de exploración. En la cinemática inversa, se restringen ángulos que no pueden ser alcanzados físicamente, o ángulos que pueden causar una colisión. las restricciones se observan en la Tabla 3.

Tabla 3.
Restricciones de las articulaciones del robot

Articulación (numerado desde la base hacia el efector final)	Límite Inferior (grados)	Límite Superior (grados)
1	-180	180
2	-80	80
3	-180	180
4	-100	100
5	-150	150
6	-100	100
7	-150	150

Fuente: [Sánchez & Terán, 2015]

2.10.2.3. Generación de Trayectorias

Una vez resuelta la cinemática inversa es necesario interpolar los puntos y hallar una trayectoria. La trayectoria dará como resultado datos de posición, velocidad y aceleración en función del tiempo.

2.10.2.4. Resultados

Luego de realizar pruebas con el robot ARD7 desarrollado por [Sánchez & Terán, 2015] se obtuvo resultados de Exactitud de posición Tabla 4 y resultados de Repetibilidad de posición Tabla 5.

Tabla 4.
Exactitud del robot AR7D

Exactitud de posición	8.8 mm
Exactitud yaw	2.63°
Exactitud pitch	2.3°
Exactitud roll	2.3°

Fuente: [Sánchez & Terán, 2015]

Tabla 5.
Repetibilidad del robot AR7D

Repetibilidad de posición	+/-1.26 mm
Repetibilidad yaw	+/-0.305°
Repetibilidad pitch	+/-0.357°
Repetibilidad roll	+/-0.357°

Fuente: [Sánchez & Terán, 2015]

2.11. Resumen

En el capítulo desarrollado, se realizó el estado del arte de los temas relacionados directamente con el Diseño e implementación del proyecto. Se abordaron conceptos básicos acerca de los manipuladores robóticos industriales. Además, se hizo una introducción a los manipuladores colaborativos, tomando información de robots duales existentes desarrollados por empresas internacionales. Se hizo referencia a la solución de cinemática directa e inversa y algunos métodos existentes para su solución. Se realizó una breve descripción para robots colaborativos y como se realiza el control de los mismos, mediante un planificador y control jerárquico. Se estudió el sensado y sistema de visión artificial en manipuladores robóticos para la detección de geometría e identificación de objetos. Por último, se realizó un resumen sobre los aspectos importantes del desarrollo del brazo robótico AR7D que será utilizado como base para el desarrollo del presente proyecto.

Capítulo 3

Metodología y Diseño Mecatrónico

En este capítulo se documenta el desarrollo del diseño en cada una de sus ramas, partiendo del estudio previo del manipulador AR7D. El diseño empieza desde el diseño mecánico, mediante la réplica del manipulador existente y se propone el diseño CAD del torso de soporte para el robot colaborativo de dos brazos. Se realizará el diseño de las conexiones eléctricas de control y potencia. Se propone el diseño de Control mediante una máquina de estados utilizando software libre para la codificación y programación.

3.1. Metodología de Diseño

La metodología utilizada en el desarrollo del diseño es el método en V definido por [Rojas, 2015], el mismo que define un procedimiento uniforme para el desarrollo del producto. Consta de tres partes: una corriente de especificación, una corriente de desarrollo y una corriente de pruebas.

El torso robótico colaborativo de dos brazos, se ha desarrollado en función del diagrama en V planteado en la Figura 34.

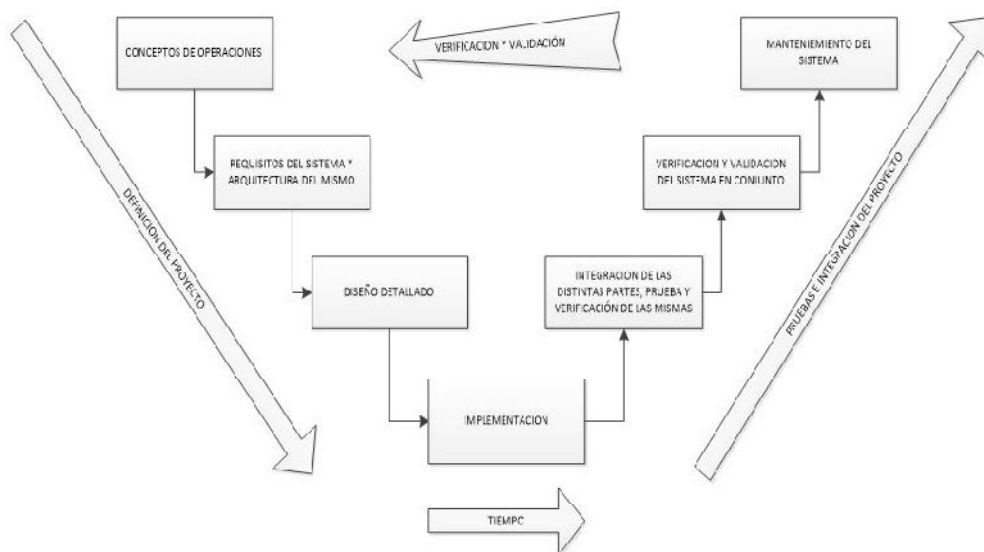


Figura 34: Metodología de desarrollo del torso - Método en V

3.1.1. Corriente de Especificación

3.1.1.1. Conceptos de Operaciones

El torso robótico colaborativo consta de un torso con dos brazos y una unidad de control llamada planificador. Desde el planificador se controla la secuencia de movimientos que realizan los brazos para su correcta ubicación frente a la placa para ser empernada. La correcta ubicación de los efectores finales se verifica y corrige por medio de un sistema de visión artificial. Una vez ubicados correctamente los brazos, el planificador envía una secuencia de pasos a los brazos para el proceso de empernado.

3.1.1.2. Requisitos y Arquitectura del Sistema

La estructura de los brazos ha sido basada en la arquitectura del brazo robótico de 7 grados de libertad realizado en la Universidad de las Fuerzas Armadas por [Sánchez & Terán, 2015]. Se ha diseñado un torso para dar soporte a los dos brazos, uno a cada lado del torso. El interior del torso contiene el cableado necesario para el envío y recepción de señales de los brazos hacia el planificador. De esta forma se maneja una arquitectura de control jerárquica entre planificador y controladores de cada brazo.

3.1.1.3. Diseño Detallado

El diseño del torso consta de cuatro partes:

1. Componentes mecánicos: consta de tres partes, diseño de base de soporte y tronco del torso, diseño del segundo brazo antropomórfico, y diseño de la celda de trabajo.
2. Componentes electrónicos/eléctricos: consta de tres partes, diseño del circuito de potencia, sensores y actuadores, y diseño del circuito de acondicionamiento.
3. Componentes del sistema de control: programación del planificador con la ayuda de dos subprocesos de control de movimiento (uno para cada brazo), posicionamiento y ejecución del proceso automático de empernado.
4. Software y hardware para implementación, análisis y simulación: se utiliza hardware libre y el Software ROS para la programación y simulación del movimiento y posicionamiento de los brazos del torso y la programación del HMI.

3.1.2. Corriente de Desarrollo

Implementación: una vez desarrollada y simulada cada etapa de diseño se procede a la implementación de las mismas.

3.1.3. Corriente de Pruebas

Integración de las diferentes partes, prueba y verificación de las mismas: una vez implementada cada etapa del diseño del torso se procederá a su integración con los brazos como un solo sistema.

Verificación y validación del sistema en conjunto: Se realiza pruebas y verificación del funcionamiento del sistema integrado.

Mantenimiento del sistema: el sistema ha sido diseñado y desarrollado de tal manera que facilite el mantenimiento del mismo de forma modular, además se ha desarrollado un manual de usuario con información acerca del mantenimiento de sus componentes.

3.1.4. Despliegue de la Función de Calidad QFD

El despliegue de la función de la calidad QFD [Díaz, 2012], mejor conocida como la voz de los clientes o casa de la calidad. La QFD es un proceso que asegura que los deseos y necesidades de los clientes sean traducidas en características técnicas.

- Definición de las necesidades: Es la primera parte de la matriz a ser desarrollada, debe considerar los requerimientos del cliente sobre el producto. Las necesidades

definen características y conceptos que se requieren para el diseño final. Las necesidades y prestaciones que se requieren sobre el Torso robótico se detallan en la Tabla 6.

Tabla 6.
Necesidades del cliente de la QFD

N°	Necesidades
1	Modular e interfaz amigable
2	Peso ligero menor a $20kg$
3	Volumen de espacio de trabajo aproximado de $1m^3$
4	Carga útil total aproximada de $3kg$
5	Alcance vertical aproximado $0,6m$
6	Alcance horizontal aproximado $1m$
7	Tamaño de la placa a empinar $0,26m * 0,16m$
8	Pernería de peso ligero para junta empinada
9	Repetibilidad aproximada
10	Material liviano y resistente

- Definición de las características técnicas: Se refiere a las características técnicas o de ingeniería del producto, en donde se reflejan las necesidades del cliente. En la Tabla 7 se detalla las características técnicas necesarias para el desarrollo del Torso para satisfacer las necesidades del cliente.

Tabla 7.
Características técnicas de la QFD

N°	Características Técnicas
1	7GDL por brazo
2	Planificador con control de brazos individual
3	Visión artificial
4	Resuelve cinemática inversa
5	Potencia suficiente para alimentación de componentes
6	Software libre
7	Controlador de servomotores
8	Componentes mecánicos impresos en 3D
9	Gripper tipo pinza
10	Control por medio de máquina de estados
11	Placa perforada
12	Servomotores
13	Estructura de base robusta - Torso apto para dos brazos
14	Interfaz gráfica HMI
15	Trabajo colaborativo y sincronizado de los brazos

La QFD utiliza un método gráfico en el que se expresan relaciones entre deseos de los clientes y las características del diseño. Es una matriz que enlista las necesidades de los clientes o también llamadas atributos comparándolas con las características de

diseño. La matriz QFD desarrollada para el torso robótico colaborativo se muestra en la Figura 35.

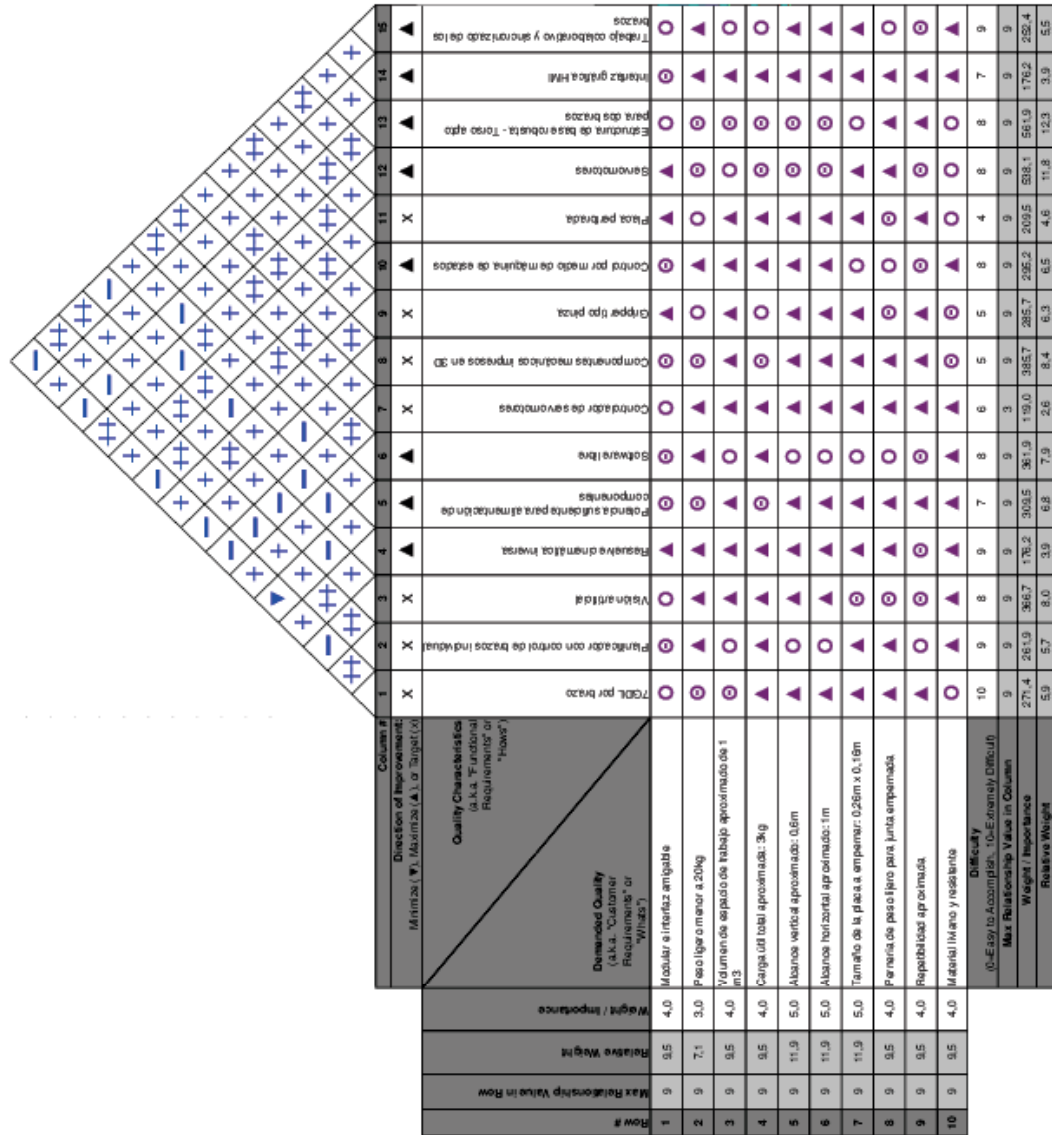


Figura 35: Casa de la calidad para el torso robótico colaborativo

Los resultados obtenidos a partir de la matriz QFD de muestran en la Tabla 8.

Tabla 8.
Resultados de la matriz QFD

Hitos	Características técnicas	Porcentaje de ponderación
1	Estructura de base robusta - Torso apto para dos brazos	12.3
2	Servomotores	11.8
3	Componentes mecánicos impresos en 3D	8.4
4	Visión artificial	8.0
5	Software libre	7.9
6	Potencia suficiente para alimentación de componentes	6.8
7	Control por medio de máquina de estados	6.5
8	Gripper tipo pinza	6.3
9	7GDL por brazo	5.9
10	Planificador con control de brazos individual	5.7
11	Trabajo colaborativo y sincronizado de los brazos	5.5
12	Placa perforada	4.6
13	Resuelve cinemática inversa	3.9
14	Interfaz gráfica HMI	3.9
15	Controlador de servomotores	2.6

3.1.5. Factibilidad y Planeación

La factibilidad del desarrollo del proyecto depende directamente del análisis de las características técnicas y la medición funcional y desempeño de cada una de estas. La Tabla 8 contiene los resultados de la matriz QFD, las características técnicas de la misma serán clasificadas en módulos según su operatividad, para formar una estructura funcional y de ésta manera cubrir las necesidades del cliente de una manera óptima. Para el desarrollo del torso robótico se establece la estructura modular de la Tabla 9.

Tabla 9.
Estructura modular del torso robótico

N°	Módulo	Subsistema	Función	Hitos a cumplir
1	Estructura	Brazos antropomórficos	Geometría y re diseño de los eslabones Movilidad Soporte de la celda de trabajo	1,2,3, 6,8,9, 12,15
		Torso robótico	Tronco de soporte de los dos brazos Puertos de conexión, botoneras Placa a empernar	
		Modificación del gripper	Realización de tareas	
2	Cinemática del manipulador	Cinemática inversa en ROS	Posicionamiento brazo 1 Posicionamiento brazo 2	13
3	Visión artificial	Sistema de detección de objetos	Verificar correcto posicionamiento TCPs	4
4	Planificador	Interfaz gráfica	Comunicación humano-máquina	5,7,10, 11,14
		Máquina de estados	Programación de tareas	

3.2. Módulo 1. Estructura

3.2.1. Subsistema 1.1. Brazos Antropomórficos

A partir del diseño mecánico del manipulador antropomórfico de siete grados de libertad desarrollado por [Sánchez & Terán, 2015] se procede a replicar el manipulador AR7D, para obtener el segundo brazo del Torso Robótico.

3.2.1.1. Diseño Geométrico

El diseño geométrico de las piezas que conforman cada eslabón y el análisis estático de las mismas fueron desarrollados por [Sánchez & Terán, 2015] y se detallan a continuación:

Base del Manipulador

Parte inferior del brazo robótico Figura 36, encargada de anclar el brazo al sitio de trabajo. El análisis estático de este elemento, considera que el elemento se encuentra sujeto mediante pernos en su parte inferior y apoyado sobre un soporte. Se aplica una carga de 2.6 Kgf sobre el eje del motor y se realiza un torque de 6 Nm sobre la brida del motor. Los resultados del análisis estático se representan en la Tabla 10.



Figura 36: Base del manipulador

Tabla 10.

Análisis estático de la base del manipulador

Pieza	Esfuerzo Von Mises	Desplazamiento	Deformación unitaria	Factor de seguridad
Base	$1.926 \times 10^8 \text{ N/m}^2$	1.22mm	6.01×10^{-1}	1.9

Eslabón 1

Eslabón rotacional Figura 37, se conecta a la base del manipulador. El análisis estático de este elemento, considera que la parte inferior está sujeta por medio de pernos al motor anterior. Soporta una carga de 2.4 kg aplicada sobre el eje del motor correspondiente al peso de los eslabones anteriores y un torque de 6 Nm en la brida del motor así como de 6 Nm producidos por el motor sobre la base del eslabón. Los resultados del análisis estático se representan en la Tabla 11.



Figura 37: Eslabón 1

Tabla 11.
Análisis estático de la base del Eslabón 1

Pieza	Esfuerzo Von Mises	Desplazamiento	Deformación unitaria	Factor de seguridad
Eslabón 1	$8.820 \times 10^7 N/m^2$	$4.247 \times 10^{-1} mm$	2.22×10^{-2}	2.57

Eslabón 2

Eslabón rotacional Figura 38, se conecta con el Eslabón 1. El análisis estático de este elemento, considera una carga de 1.647 Kgf sobre el eje del motor correspondiente a los eslabones que siguen en la cadena cinemática, esta carga también produce un torque de 3.4 Nm en la brida del motor. El motor anterior del Eslabón 2 y el que se sujeta al Eslabón 3 son del tipo MX-64T, cada uno produce un torque máximo de 6 Nm. Los resultados del análisis estático se representan en la Tabla 12.



Figura 38: Eslabón 2

Tabla 12.
Análisis estático de la base del Eslabón 2

Pieza	Esfuerzo Von Mises	Desplazamiento	Deformación unitaria	Factor de seguridad
Eslabón 2	$2.16 \times 10^8 N/m^2$	$1.873 \times 10^{-1} mm$	2.21×10^{-2}	1.1

Eslabón 3

Eslabón rotacional Figura 39, se conecta con el Eslabón 2. El análisis estático realizado por [Sánchez & Terán, 2015] de este elemento, considera que sobre el eje del motor se apoyan los siguientes eslabones, motores y actuador final con un peso

total de 1.462 kg. El torque máximo que puede ejercer el motor MX-28T es de 2.5 Nm. Los resultados del análisis estático se representan en la Tabla 13.



Figura 39: Eslabón 3

Tabla 13.
Análisis estático del Eslabón 3

Pieza	Esfuerzo Von Mises	Desplazamiento	Deformación unitaria	Factor de seguridad
Eslabón 3	$1.354 \times 10^8 N/m^2$	$8.452 \times 10^{-1} mm$	4.690×10^{-2}	3.732

Eslabón 4

Eslabón rotacional Figura 40, se conecta con el Eslabón 3. El análisis estático de este elemento, considera la aplicación de una carga en el eje del motor, el mismo que soporta la carga y el valor es el peso de los eslabones posteriores más la carga aplicada ya considerada en el efector final, esto da una carga de 1.355 kg. Además, se considera un momento en el eje del motor con el valor del torque del motor que es de 1.6 Nm, el torque que ejerce el motor anterior MX-28T es de 2.5 Nm y el torque que se produce por consecuencia del peso de los elementos siguientes y la carga aplicada que resulta de 2.464 Nm. Los resultados del análisis estático se representan en la Tabla 14.



Figura 40: Eslabón 4

Tabla 14.
Análisis estático del Eslabón 4

Pieza	Esfuerzo Von Mises	Desplazamiento	Deformación unitaria	Factor de seguridad
Eslabón 4	$1.77 \times 10^7 N/m^2$	$10.96 \times 10^{-1} mm$	5.791×10^{-3}	1.694

Eslabón 5

Eslabón rotacional Figura 41, se conecta con el Eslabón 4. El análisis estático de este elemento, considera la aplicación de cargas en el extremo así como el peso del ensamblaje. En el eje del motor se aplica el peso de los eslabones posteriores más la carga aplicada ya considerada en el efector final, esto da una carga de 1.236 kg, adicionalmente se aplica un momento en el eje del motor con el valor del torque del motor que es 1.6Nm. Considerando también el torque que ejerce el motor anterior AX-12A de 1.6 Nm, el torque que se produce por consecuencia del peso de los elementos siguientes y la carga aplicada resulta 1.71 Nm. Los resultados del análisis estático se representan en la Tabla 15.



Figura 41: Eslabón 5

Tabla 15.
Análisis estático del Eslabón 5

Pieza	Esfuerzo Von Mises	Desplazamiento	Deformación unitaria	Factor de seguridad
Eslabón 5	$6.773 \times 10^6 N/m^2$	$2.51 \times 10^{-1} mm$	6.682×10^{-3}	1.496

Eslabón 6

Eslabón rotacional Figura 42, se conecta con el Eslabón 5. El análisis estático de este elemento, considera la aplicación de cargas en el extremo, así como el peso del ensamblaje. El eje del motor soporta el peso de los eslabones posteriores más la carga aplicada ya considerada en el efector final, esto da una carga de 1.1 kg. Además, se aplica un momento en el eje del motor con el valor del torque de dicho motor, consideramos también el torque que ejerce el motor anterior AX-12A es de 1.6 Nm, el torque que se produce por consecuencia del peso de los elementos siguientes y la carga aplicada resulta 0.78815 Nm. Los resultados del análisis estático se representan en la Tabla 16.



Figura 42: Eslabón 6

Tabla 16.
Análisis estático del Eslabón 6

Pieza	Esfuerzo Von Mises	Desplazamiento	Deformación unitaria	Factor de seguridad
Eslabón 6	$2.005 \times 10^7 N/m^2$	$7.27 \times 10^{-1} mm$	4.272×10^{-3}	2.264

Soporte Gripper

Eslabón rotacional Figura 43, une al manipulador con el efector final. El análisis estático de este elemento, considera la aplicación de cargas en el extremo así como

el peso del ensamblaje. El eje del motor soporta el peso de los eslabones posteriores más la carga aplicada ya considerada en el efector final, esto da una carga de 1.043 kg. Adicionalmente, se aplica un momento en el eje del motor con el valor del torque de dicho motor, se considera también el torque que ejerce el motor anterior AX-12A es de 1.6 Nm, el torque que se produce por consecuencia del peso de los elementos siguientes y la carga aplicada resulta 0.024 Nm. Los resultados del análisis estático se representan en la Tabla 17.



Figura 43: Soporte de Gripper

Tabla 17.
Análisis estático del Soporte del Gripper

Pieza	Esfuerzo Von Mises	Desplazamiento	Deformación unitaria	Factor de seguridad
Soporte Gripper	$4.23 \times 10^6 N/m^2$	$1.453 \times 10^{-1} mm$	1.506×10^{-3}	7.083

Efector Final tipo Gripper

El efector final de tipo pinza se muestra en la Figura 44. En donse se puede observar la geometría del mismo acoplado al último eslabón

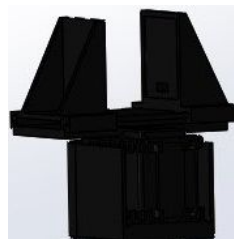


Figura 44: Efector final tipo gripper

Una vez ensamblados los eslabones, se obtiene como resultado final un segundo brazo antropomorfo con la geometría de un robot manipulador de siete grados de libertad como se muestra en la Figura 45.

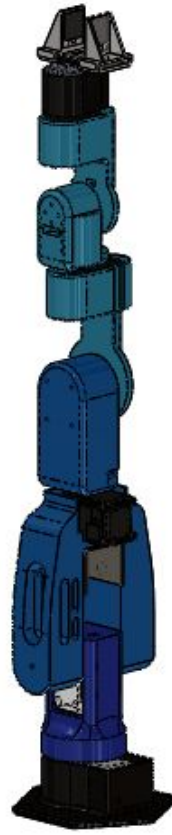


Figura 45: Geometría del robot manipulador

3.2.1.2. Movilidad de los Eslabones

De acuerdo al diseño propuesto por [Sánchez & Terán, 2015] se seleccionaron los servomotores Dynamixel de tres modelos diferentes AX-12a, MX-28T y MX-64T, cuyas características constan en la Tabla 18.

Tabla 18.
Motores Dynamixel

Especificaciones	AX-12A	MX-28T/MX-28R	MX-64T/MX-64R
Peso	54.6 g	72 g	126 g
Stall torque	1.5 Nm/16.5 Kgcm	3.1 Nm/31.6 Kgcm	7.3 Nm/74.4 Kgcm
Velocidad	59 rpm	97 rpm	78 rpm
Tensión de funcionamiento	9-12 VDC	10-14.8 VDC	10-14.8 VDC
Consumo de corriente	1.5 A	1.3 A - 1.7 A	3.9 A - 5.2 A
Dimensiones	32 mm x 50 mm x 40 mm	35.6 mm x 50.6 mm x 35.5 mm	40.2 mm x 61.1 mm x 41 mm
Ángulo	300°	360°	360°
Reducción engranes	254:1	193:1	200:1
Material de tren de engranajes	Plástico	Acero endurecido	Acero endurecido
Sensor de Posición	Potenciómetro	Codificador magnético	Codificador magnético
Protocolo de comunicación	TTL	TTL/RS-485	TTL/RS-485
Velocidad de comunicación	1 Mbps	3 Mbps	3 Mbps
Cumplimiento/PID	Cumplimiento	PID	PID
Ilustración			

Fuente: [TrossenRobotics, 2015]

Alimentación de los Servomotores

En la Tabla 19 se detalla la corriente requerida para el funcionamiento de los brazos del torso.

Tabla 19.
Corriente máxima de consumo de los motores Dynamixel a 12 VDC

Modelo	Corriente	Nº motores por brazo
MX-64T	4.1 A	3
MX-28T	1.4 A	1
AX-12A	1.5 A	4

La corriente necesaria para el funcionamiento del torso es de 4.1A siendo el consumo máximo del servomotor MX-64T al torque máximo. Se dispone el uso de una fuente de poder con una salida de 12 VDC a 6 A.

Controlador USB2Dynamixel

El controlador USB2Dynamixel [TrossenRobotics.com, 2016], es un controlador capaz de controlar cualquier red de actuadores Dynamixel conectados de forma serial, desde un puerto USB o Serial desde un ordenador como se muestra en la Figura 46.

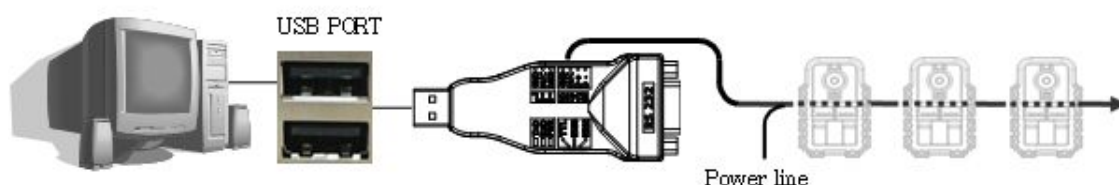


Figura 46: Control de Dynamixel desde PC
Fuente:[TrossenRobotics.com, 2016]

El controlador se puede programar con librerías de soporte en entornos como Matlab, Microsoft Visual Studio, Java, C, C++, Visual Basic, Python y Labview. El adaptador del controlador USB2Dynamixel [TrossenRobotics.com, 2016] tiene tres puertos de comunicación, además del puerto USB y se los puede visualizar en la Figura 47:

- TTL: conector de 3 pines utilizados con las serie Dynamixel AX. Los actuadores AX-12A, AX-18 y el módulo sensor AX-S1 se conectan a este puerto.
- RS485: conector de 4 pines, utilizados en los actuadores Dynamixel de altas prestaciones DX, RX y EX.
- RS232 DB9: conector serie estándar RS232 de 9 pines. Este puerto es ideal para conectar el adaptador ZIG2Serial.

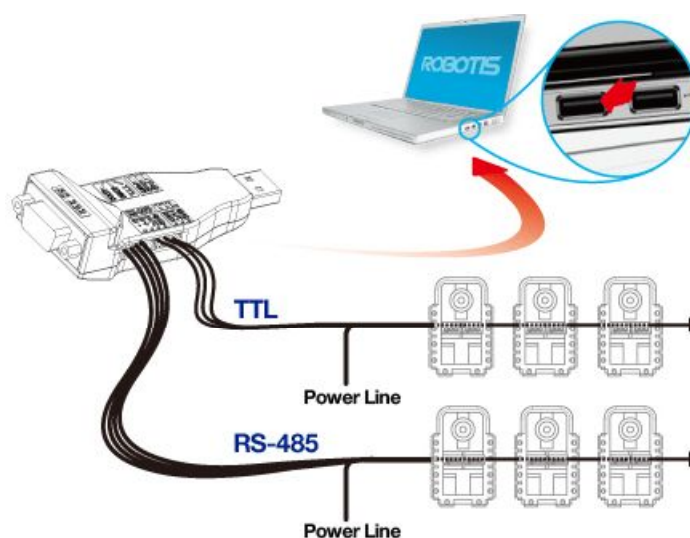


Figura 47: Puertos de comunicación del Controlador USB2Dynamixel
Fuente: [TrossenRobotics.com, 2016]

3.2.1.3. Esquema de Conexión

El torso robótico utiliza dieciséis motores Dynamixel, ocho en cada brazo, los mismos que se conectan de manera serial al controlador USB2Dynamixel por medio del puerto de comunicación TTL y se alimentan por medio de una fuente de alimentación de 12 VDC - 6 A como se muestra en la Figura 48.

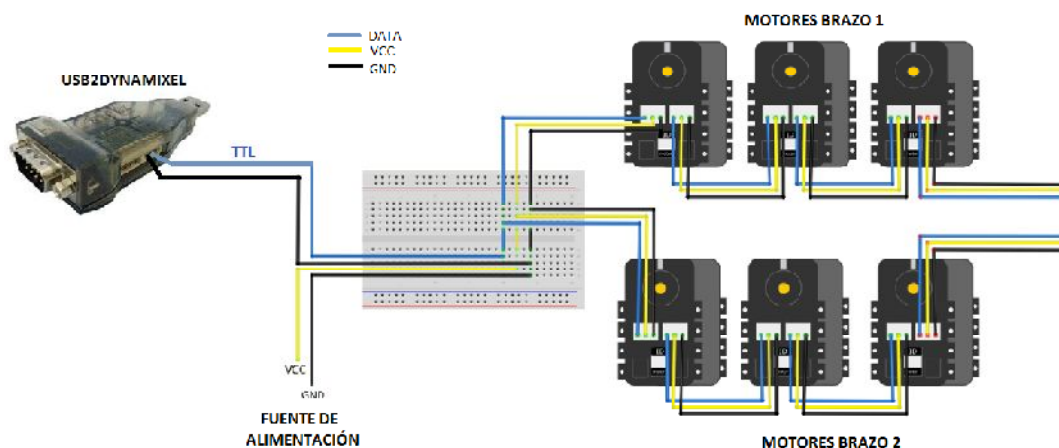


Figura 48: Esquema de conexión Torso Robótico

3.2.1.4. Configuración de los Motores Dynamixel

Para realizar la configuración y verificar el correcto funcionamiento de cada uno de los motores Dynamixel, se necesita del software RoboPLUS disponible en la página electrónica de Robotis. El software dispone de una opción llamada Dynamixel Wizard

que es la opción que se necesita para la configuración. El proceso de configuración de motores en el programa RoboPlus se lo detalla en el flujograma de la Figura 49.

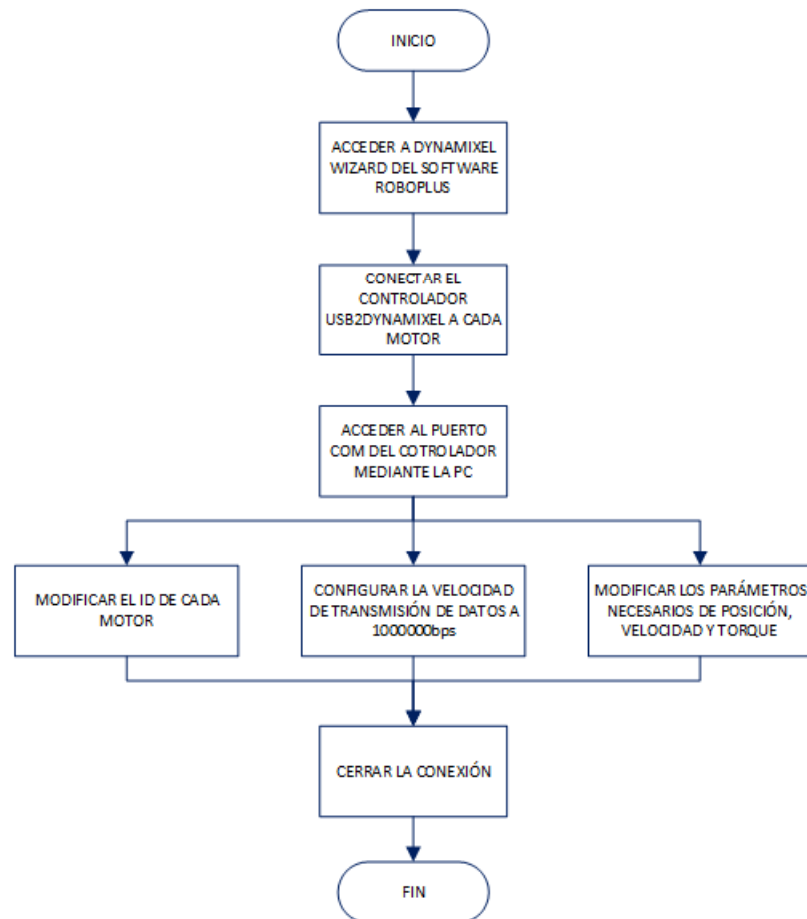


Figura 49: Diagrama de configuración de motores en Wizard

Una vez identificados todos los motores conectados en serie como se observa en la Figura 50, se puede acceder a cada uno de ellos y realizar las modificaciones necesarias.

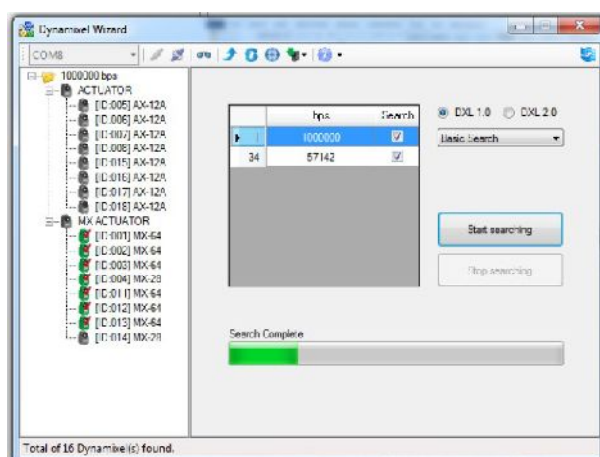


Figura 50: Búsqueda e identificación de los motores conectados en serie

3.2.1.5. Manipulación de los Motores Dynamixel desde Ubuntu

Para manipular cada uno de los motores Dynamixel desde Ubuntu es necesario instalar Herborist. El programa Herborist corre en el paquete PyPot para el cual es necesario realizar la instalación de varios paquetes adicionales. Desde la página poppy-project.github.io se puede acceder a los links de instalación respectivos como se observa en la Figura 51.

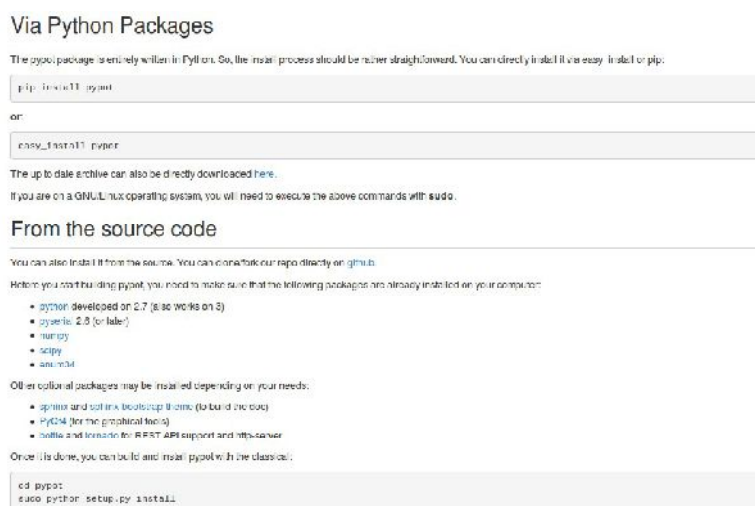


Figura 51: Paquetes de instalación de PyPot y programas adicionales

Una vez instalados todos los paquetes requeridos, ellos constarán entre las carpetas como se observa en la Figura 52. Los paquetes necesarios para utilizar PyPot son Python 2.7 o posterior, Pyserial, Numpy, Scipy y Enum34.

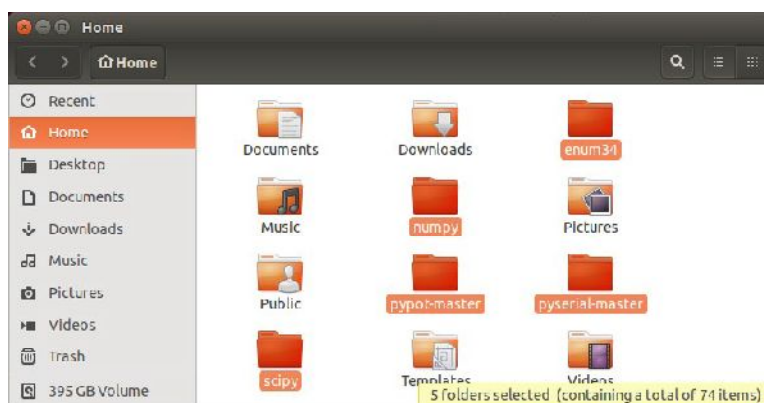


Figura 52: Paquetes requeridos instalados en Ubuntu

Si el paquete de PyPot se ha instalado correctamente, es posible ejecutar el programa Herborist desde el Terminal como se observa en la Figura 53.



Figura 53: Ejecución del programa Herborist desde el Terminal

Al ejecutar el programa se abrirá el espacio de trabajo del mismo, de forma que se puede acceder a la manipulación de los motores como se observa en la Figura 54. En este programa podemos seleccionar el rango de comunicación de los motores que en este caso están configurados todos con 1000000 y proceder a presionar scan para identificar los motores conectados a la PC. Al terminar la búsqueda se puede acceder a cada uno de los motores y manipularlos mediante las opciones que proporciona el programa Herborist.

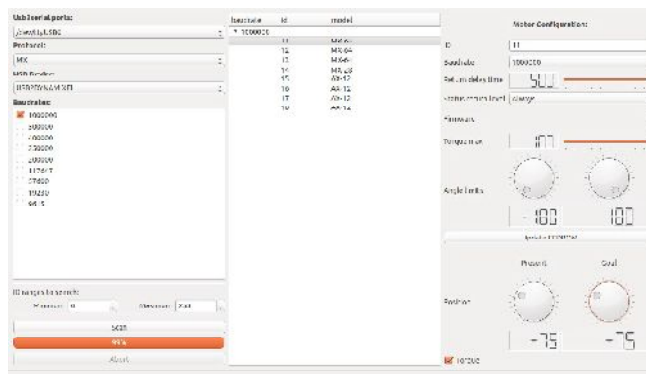


Figura 54: Espacio de trabajo del programa Herborist

Cabe recalcar que en el programa Herborist para Ubuntu tiene limitaciones en

cuanto a los datos de los motores a los cuales se puede acceder en comparación a Dynamixel Wizard disponible en Windows.

3.2.2. Subsistema 1.2. Torso Robótico

Para diseñar la estructura del torso robótico se partió del diseño y tamaño del AR7D y se propuso dos dibujos preliminares que se detallan a continuación:

Diseño 1:

Se propone el Diseño 1, consta de una estructura cubica de dos partes. La parte inferior con dos patas laterales que permitirán anclar al torso a una base fija, y una parte superior con 6 perforaciones a cada lado del cubo para la unión del eslabón base del brazo robótico, y un orificio en la parte inferior de las perforaciones para el ingreso del cableado. El cubo contendrá las fuentes, el conexionado de los brazos, el controlador USB2Dynamixel. En la parte posterior se propone un panel de control con botoneras on/off, paro de emergencia, una luz de confirmación del funcionamiento del sistema y los puertos de conexión de las fuentes de poder y puerto usb para la comunicación del controlador hacia el ordenador. El Diseño 1 se muestra en la Figura 55.



Figura 55: Diseño 1

Diseño 2:

Se propone el Diseño 2 que consta de una base cuadrada con 4 perforaciones una en cada esquina, que permitirán anclar el torso a una base fija. Sobre la base cuadrada se propone la implementación de un cubo con una abertura en la parte posterior, dentro de este cubo se implementará las fuentes, el conexionado de los brazos, el conexionado de botoneras y el controlador USB2Dynamixel. La tapa del cubo tendrá orificios para los puertos de conexión de las fuentes de poder, un puerto USB para la comunicación

del controlador USB2Dynamixel al ordenador, un botón de on/off del sistema, una luz de confirmación del equipo en funcionamiento y un paro de emergencia.

Se propone el diseño del tronco de la estructura por medio de dos figuras cilíndricas, en el cilindro superior horizontal se diseña dos tapas, una a cada lado con 6 perforaciones para la unión del eslabón base del brazo robótico, y un orificio en la parte inferior para el ingreso del cableado de los motores, como se puede observar en la Figura 56.



Figura 56: Diseño 2

Para realizar el diseño del torso robótico del manipulador colaborativo de dos brazos, se tomaron en cuenta varios factores como la altura, facilidad de construcción, adquisición de materiales y complejidad de manufactura. Para identificar los factores más importantes y elegir la mejor opción de diseño se realizó la Tabla 20 comparativa de dos diseños iniciales.

Tabla 20.

Tabla comparativa de diseños preliminares

Diseño		Diseño 1	Diseño 2
Tamaño estructura	Altura	378mm	643,71mm
	Ancho	500mm	300mm
	Profundidad	350mm	300mm
Volumen de trabajo		0,296m ³	0,961m ³
Funcionalidad		Medio	Alta
Facilidad de construcción		Media	Media
Material		Acero	Acero

Debido a la facilidad de construcción, el mayor tamaño, y volumen de trabajo que dan estabilidad y mayor funcionalidad al torso, se decidió desarrollar e implementar el Diseño 2.

3.2.2.1. Desarrollo del Diseño

Se realizó el diseño de la base del manipulador en función de varios aspectos. Los requerimientos de diseño en este caso fueron el espacio necesario para los componentes electrónicos, tanto para las fuentes de alimentación como para la tarjeta de adquisición y demás elementos. Adicionalmente, se tomó en cuenta el material de construcción y algunas de sus características mecánicas. Se dejó una cara libre para colocar una placa de acrílico como puerta de acceso de mantenimiento como se puede observar en la Figura 57.

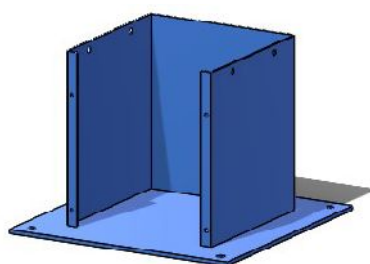


Figura 57: Diseño de la base del torso

Para proceder al diseño del cuerpo del manipulador se tomó en cuenta el acople a la base, la altura requerida para evitar la limitación del espacio de trabajo de los brazos del manipulador colaborativo y el espacio disponible tanto para el cableado como para el acople de ambos brazos antropomorfos. Una vez identificados los requerimientos de diseño del torso se procedió a realizar el diseño en forma de cilindro como se observa en la Figura 58 y el diseño de las tapas del torso que contendrán los brazos del robot como se observa en la Figura 59. De esta forma se logró, mediante software CAD, determinar las dimensiones, geometría y distribución correcta de las diferentes partes del cuerpo del manipulador.

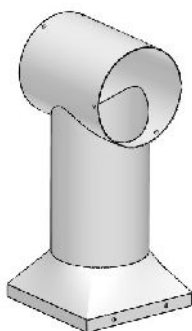


Figura 58: Torso

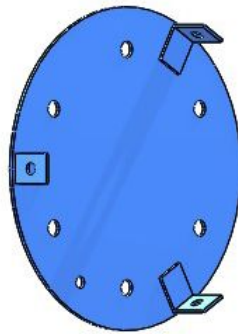


Figura 59: Tapas del torso

El ensamble del diseño de cada una de las partes del torso se observa en la Figura 60. En la imagen se puede identificar los componentes diseñados que constituyen los cilindros vertical y horizontal, las tapas laterales para empernar los brazos antropomorfos, la transición y la base del torso.



Figura 60: Estructura mecánica del Torso robótico

Una vez realizado el ensamble de las partes diseñadas junto a los dos brazos antropomorfos, se consigue el diseño final como se observa en la Figura 61.

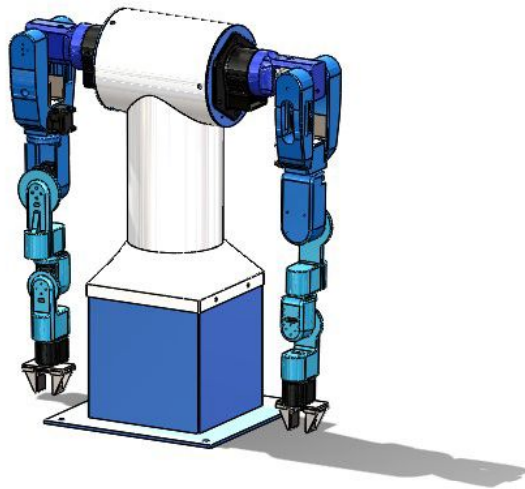


Figura 61: Diseño final de manipulador colaborativo de dos brazos

Los materiales seleccionados para la implementación del diseño realizado fueron:

- Tol de 1 mm de espesor para el torso.
- Tol de 2 mm de espesor para la base del manipulador y las tapas de soporte de ambos brazos a la estructura.
- Plancha de 6 mm de espesor para la estructura base de soporte

De acuerdo al diseño y los materiales seleccionados, se procedió a realizar el estudio estático y simulación de esfuerzos en los elementos críticos del diseño planteado.

3.2.2.2. Análisis Estático

El material seleccionado para el diseño del torso robótico fue acero ASTM A36. Los elementos identificados como partes críticas debido a la carga que deben soportar una vez implementados fueron la base del manipulador y las tapas del torso en donde se juntan ambos brazos al cuerpo del robot.

Consideraciones para el análisis estático de los elementos:

- Peso de cada brazo: 20 N
- Peso máximo de carga por brazo: 10 N
- Peso de la estructura del torso: 30 N

Análisis de Volcamiento en la Estructura

Para obtener el peso que debe tener la estructura para que sea capaz de soportar una fuerza $F = 3kg$ (se asume que cada brazo carga un máximo de $1.5kg$), se realiza un análisis de volcamiento para la estructura. Para lo cual se realiza el cálculo de un Momento Volcador el cuál se encuentra dado por la fuerza máxima que soporta la estructura ($F = 3kg$) y la distancia medida desde el centro de gravedad hasta el punto de aplicación de la Fuerza $dy = 280mm$, y el cálculo del momento estabilizador que es el peso propio de la estructura P y la distancia entre su recta de acción $dx = 100,59mm$.

Una vez obtenidos los dos momentos y tomando un factor de seguridad para el volcamiento $fs = 1,5$, se procede a encontrar el peso de la estructura.

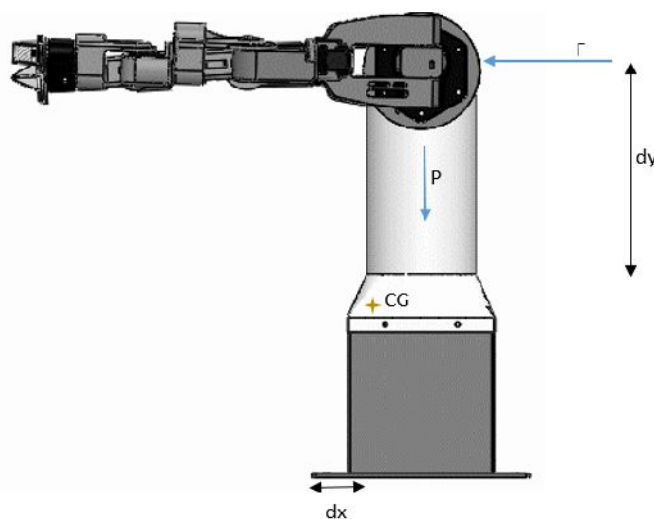


Figura 62: Análisis de volcamiento

En la Figura 62 se muestra el diagrama de Fuerzas que actúan en la estructura.

Cálculos:

- Momento volcador:

$$Mv = F(g)(dy) \quad (3.1)$$

- Momento estabilizador:

$$Me = P(g)(dx) \quad (3.2)$$

- Relación de momentos para el cálculo de factor de seguridad fs :

$$\frac{Me}{Mv} \geq fs \quad (3.3)$$

- Se reemplaza la Ecuación 3.1 y la Ecuación 3.2 en Ecuación 3.3:

$$\frac{P(g)(dx)}{F(g)(dy)} = fs \quad (3.4)$$

- Despeje de la Ecuación 3.4 para obtener P:

$$P = \frac{fs(F)(dy)}{dx} \quad (3.5)$$

$$P = \frac{1,5(3)(280)}{100,59} \quad (3.6)$$

De donde se obtiene un $P = 12,52kg$.

El diseño realizado tiene un $P = 13,82kg$, obteniendo un $fs = 1,66$, siendo este mayor a 1,5, evitando de esta manera el volcamiento de la estructura.

Análisis Estático en los Soportes de los Brazos del Torso

La verificación estructural de los soportes de los brazos del torso, llamados tapas del torso, se ponen a prueba mediante un análisis estático en el software CAD Solid-Works. El software formula desplazamientos de elementos finitos de cargas internas y externas, para calcular desplazamientos, tensiones y deformaciones de los elementos.

Para realizar el análisis estático de las tapas del torso del manipulador se tomó en cuenta un peso aproximado de 2 kg. Además, se tomó como carga máxima un peso de 1 kg por brazo, tomando en cuenta el detalle de diseño realizado por Sánchez y Terán. Se procedió a realizar el estudio estático del elemento bajo las condiciones señaladas obteniendo como resultados los mostrados en la Tabla 21, en la Tabla 22, en la Tabla 23 y en la Tabla 24.

Tabla 21.
Análisis de Tensiones en las tapas del Torso

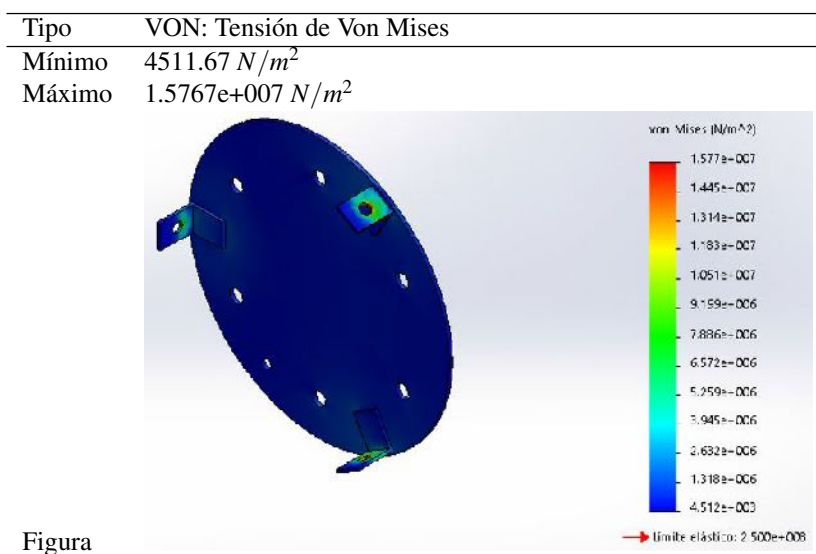


Tabla 22.
Análisis de desplazamiento en las tapas del Torso

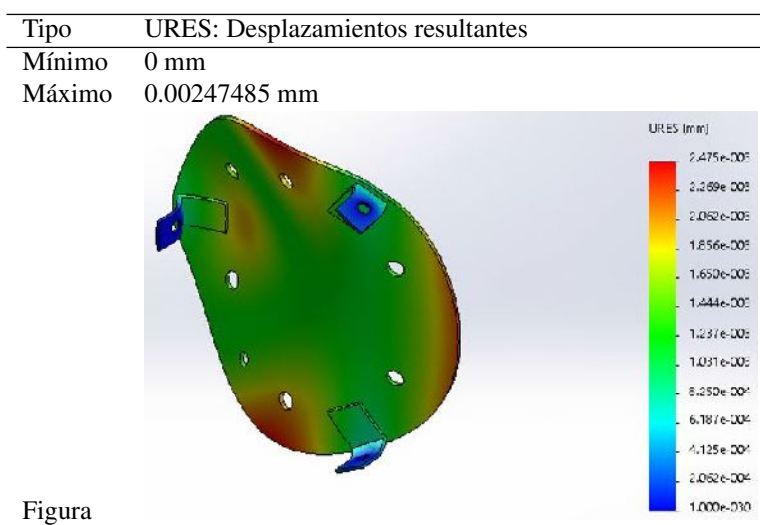
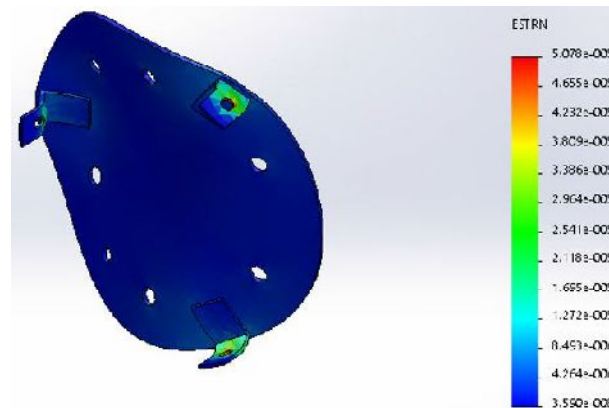


Tabla 23.
Análisis de deformaciones unitarias en las tapas del Torso

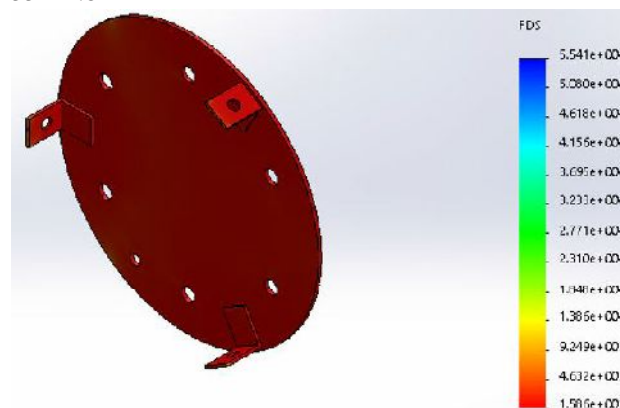
Tipo	ESTRN: Deformación unitaria equivalente
Mínimo	3.58963e-008
Máximo	5.07781e-005



Figura

Tabla 24.
Análisis de Factor de Seguridad en las tapas del Torso

Tipo	Automático
Mínimo	15.8558
Máximo	55411.8



Figura

De acuerdo a los resultados obtenidos mediante la simulación del torso del manipulador realizada con la ayuda del software CAD se observa que los desplazamientos y deformaciones existentes después de ejercer carga sobre cada una de las tapas emperradas a los brazos antropomorfos son despreciables. El factor de seguridad resultante de la simulación es superior a 1. En conclusión, el diseño mecánico propuesto para el torso del manipulador cumple con los requerimientos planteados.

Aplastamiento en la Base del Torso.

Se realiza el análisis de aplastamiento en las 4 perforaciones de la base del torso, perforaciones correspondientes al ensamble de la base con la estructura del torso. La carga que soportan las perforaciones es de 90 N.

Calculo de Aplastamiento en las Perforaciones

Para el cálculo del aplastamiento se requiere conocer el ancho de la placa $a = 200mm$, el espesor de la placa $e = 2mm$ y el diámetro de las perforaciones $d = 7mm$, dimensiones que se representan en la Figura 63.

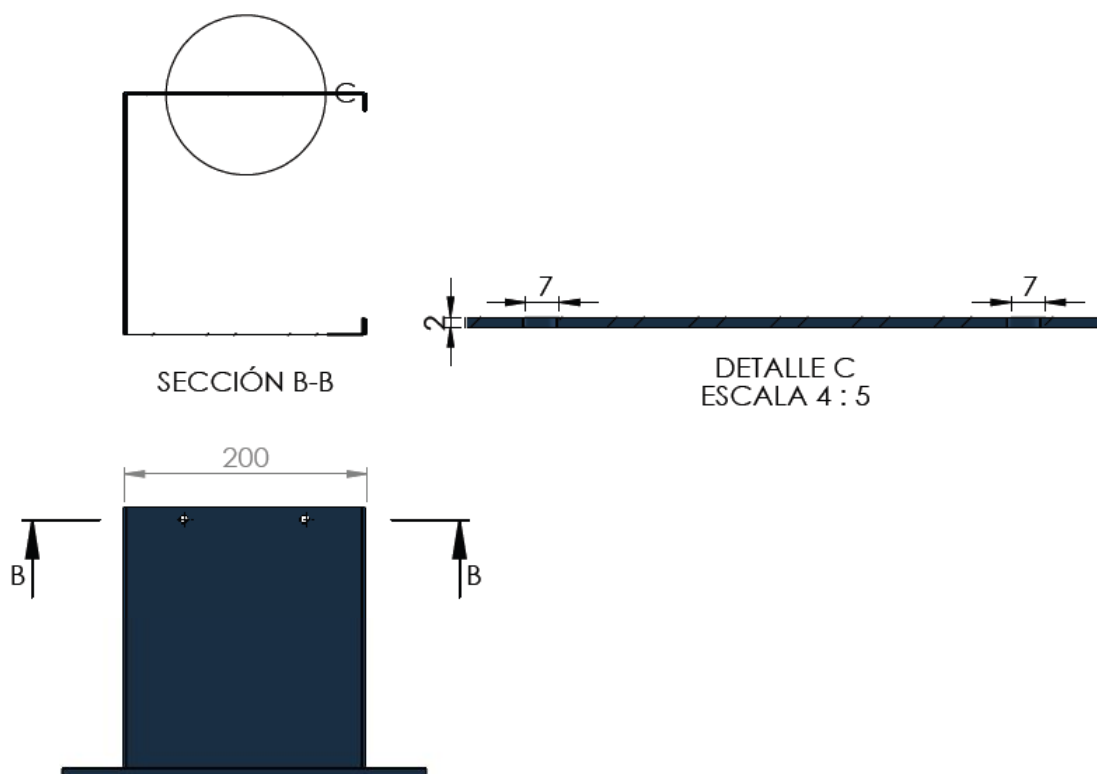


Figura 63: Dimensiones para el cálculo de aplastamiento en agujeros de la base del Manipulador.

- Para calcular el área de análisis, se reemplazan los valores de ancho a , espesor e y diámetro d en la siguiente ecuación:

$$A = e(a - 2d) \quad (3.7)$$

$$A = 2(200 - 2(7)) \quad (3.8)$$

Obteniendo así un área de: $A = 372mm^2$

- Para el cálculo del esfuerzo de trabajo, es necesario conocer la carga que se aplica en las perforaciones, para lo cual se divide la carga total para cada lado de la base en donde se encuentran las perforaciones, obteniendo una carga $P = 45N$.

$$\sigma = \frac{P}{A} \quad (3.9)$$

$$\sigma = \frac{45}{372} \quad (3.10)$$

Obteniendo así un esfuerzo de trabajo de $\sigma = 0,12096Pa$

Cálculo del Factor de Seguridad

El factor de seguridad se obtiene del esfuerzo máximo del material y el esfuerzo de trabajo, de la expresión:

$$F_s = \frac{\sigma_{mat}}{\sigma} \quad (3.11)$$

Se tiene que el esfuerzo del material es $\sigma_{mat} = 250 \times 10^6 Pa$ y el σ de trabajo obtenido en la ecuación 3.10 se tiene:

$$F_s = \frac{250 \times 10^6}{0,12096} \quad (3.12)$$

De donde el factor de seguridad es $F_s = 2,066 \times 10^9$, siendo $F_s > 1$ se comprueba que la base del torso soporta el aplastamiento producido por el peso del torso y los brazos.

Análisis Estático en la Base del Manipulador

Para realizar el análisis estático de la base del manipulador se tomó en cuenta, además del peso y la carga máxima de cada brazo, el peso de la estructura del torso diseñada. El peso resultante del diseño en software CAD fue aproximado a 3kg por lo que se procedió a realizar el análisis de carga en la base con un peso total de 90 N. Siendo los resultados obtenidos de la simulación los que se observan en la Tabla 25, en la Tabla 26, en la Tabla 27 y en la Tabla 28.

Tabla 25.
Análisis de Tensiones de la Base del Manipulador

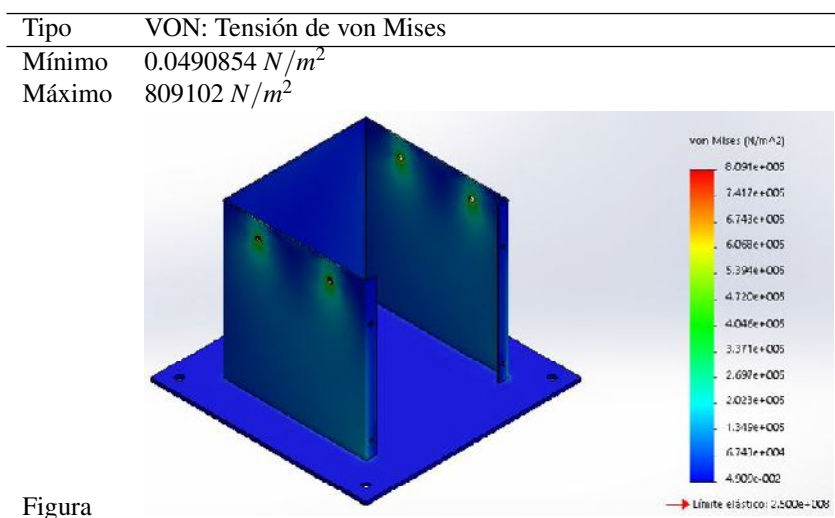


Tabla 26.
Análisis de desplazamiento de la base del Manipulador

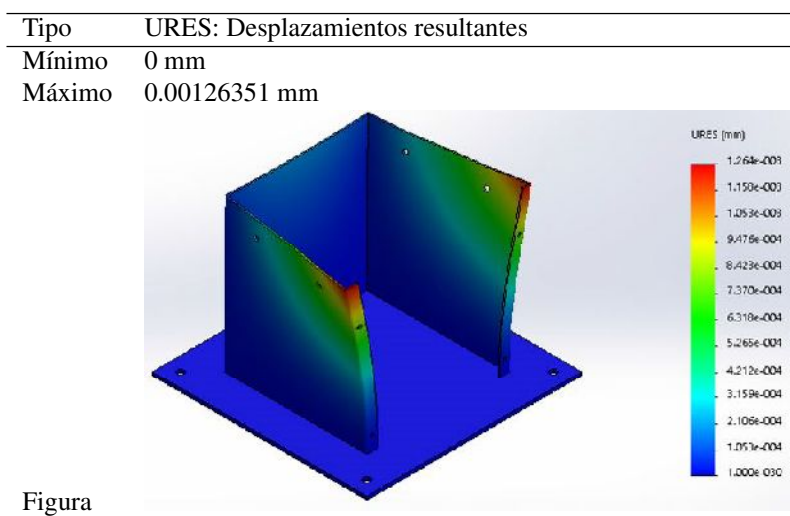
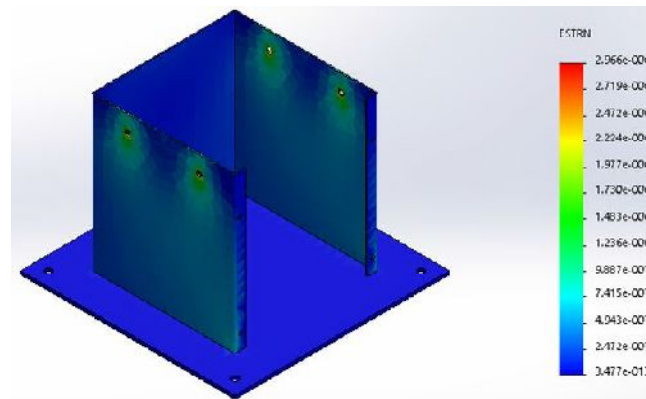


Tabla 27.
Análisis de deformaciones unitarias de la base del Manipulador

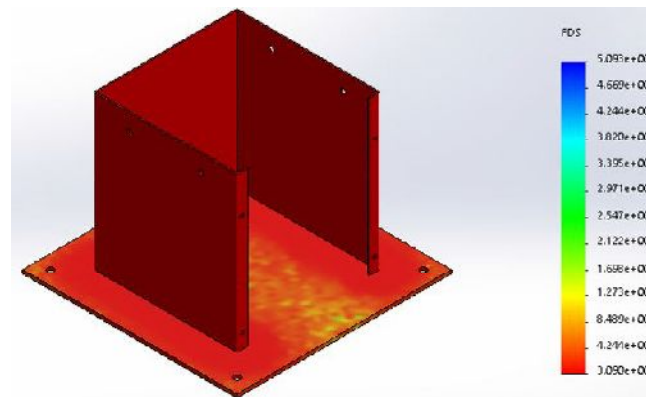
Tipo	ESTRN: Deformación unitaria equivalente
Mínimo	3.47733e-013
Máximo	4.21523e-006



Figura

Tabla 28.
Análisis de Factor de Seguridad de la base del Manipulador

Tipo	Automático
Mínimo	308.984
Máximo	5.09317e+009



Figura

De acuerdo a los resultados obtenidos mediante la simulación de la base del manipulador realizada con la ayuda del software CAD se observa que los desplazamientos y deformaciones existentes después de ejercer carga sobre el elemento de interés son despreciables. El factor de seguridad resultante de la simulación es superior a 1. En conclusión, el diseño mecánico propuesto para la base del manipulador cumple con los requerimientos planteados. Por lo tanto, se procede a la construcción del diseño propuesto.

Justificación Fs

Debido a las especificaciones necesarias en cuanto a la parte estructural del sistema mecánico del manipulador, se realizó un análisis estático de cada uno de sus componentes y del sistema completo. El diseño del torso robótico fue realizado en función al peso y la aplicación final de cada componente. Una vez realizado el análisis estático mediante elementos finitos de los componentes, siendo estos: el torso, la base y las tapas laterales del manipulador, se obtuvieron valores de factor de seguridad que superan el orden de los mil. Sin embargo, al realizar el análisis de volcamiento, tomando en cuenta el factor de seguridad implicado de 1.5, el peso de la estructura final supera este valor. El diseño desarrollado para la implementación de la estructura del manipulador cumple con las especificaciones requeridas.

3.2.3. Subsistema 1.3. Modificación del Gripper

Se realizó el diseño del gripper en función de las medidas y características tanto de la cámara Logitech utilizada para implementar visión artificial, como del eslabón del efector final de cada brazo del manipulador. El diseño del acople que se observa en la Figura 64 se desarrolló en software CAD. El plano del mismo se encuentra como anexo al final del documento en el apéndice B17.



Figura 64: Acople para cámara

3.3. Módulo 2. Cinemática del Manipulador

3.3.1. Subsistema 2.1. Cinemática Inversa en ROS

La resolución de la cinemática inversa del manipulador se realizó en base al uso de algoritmos genéticos en Python. El método utilizado para resolver el cálculo de las posiciones que debe alcanzar cada motor se lo realizó en función al diagrama de flujo de la Figura 65.

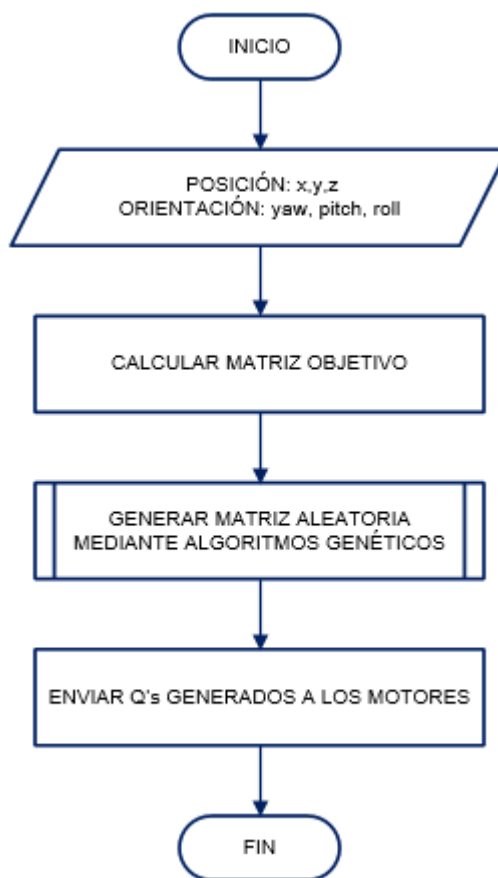


Figura 65: Cinemática inversa

Para desarrollar el cálculo se necesita el ingreso de posición x , y , z y orientación yaw, pitch y roll. Con los datos ingresados se procede a realizar el cálculo de la matriz teórica u objetivo mediante la conversión de los datos a una matriz transformada. Mediante algoritmos genéticos se genera una matriz transformada a partir de valores aleatorios que se compara con la matriz objetivo para cumplir con las posiciones ingresadas. Al finalizar el cálculo se obtienen los valores Q 's o posiciones de los motores para ejecutar el posicionamiento del brazo.

Para entender el procedimiento que desarrolla el algoritmo genético en la resolu-

ción se sigue el diagrama de flujo del subproceso para generar la matriz aleatoria que se muestra en la Figura 66.

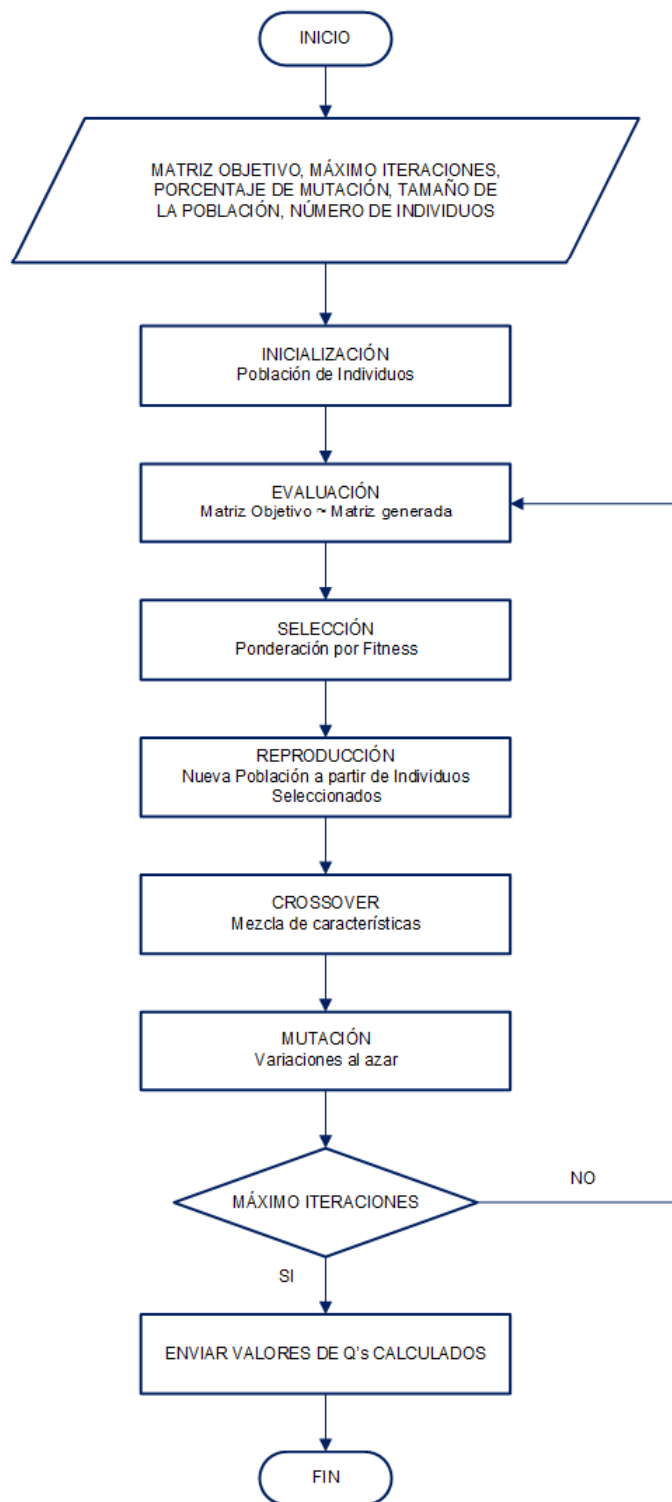


Figura 66: Algoritmos genéticos

El cálculo de las posiciones necesarias para enviar a los motores se calcula median-

te la generación de una matriz aleatoria. El algoritmo genético inicializa una población de posibles valores aleatorios para obtener una matriz generada. Posteriormente evalúa y compara cada uno de los valores de la Matriz Objetivo con la Matriz Generada. Selecciona los valores más aproximados y los reproduce para crear una nueva población a partir de ellos. Se hace una mezcla de las características de la nueva población y se ingresan valores al azar mediante mutación. El proceso se repite hasta cumplir con el número máximo de iteraciones. Al finalizar el proceso se envían los valores calculados de las posiciones requeridas para los motores.

Mediante el uso de algoritmos genéticos se logró resolver la cinemática inversa de cada uno de los manipuladores con un error cuadrático medio menor a 0.1. Los resultados obtenidos a partir del método seleccionado para la resolución de la matriz de transformación cumplen con las especificaciones de posicionamiento teóricas.

3.4. Módulo 3. Visión Artificial

3.4.1. Subsistema 3.1. Sistema de Detección de Objetos

El sistema de detección de objetos, fue desarrollado con el uso del paquete OPENCV (Open Source Computer Vision) de Python, mediante un programa de detección de colores. El diagrama de la Figura 67 muestra el flujo del sistema de visión artificial implementado. Al activarse el sistema de visión artificial se encienden dos cámaras, una en cada efector final. La cámara uno se encuentra programada para detectar la presencia de un perno y la cámara dos se encuentra programada para detectar la presencia de una tuerca, si no se detectan los dos elementos el programa enviará un mensaje a la interfaz gráfica indicando que “NO EXISTEN ELEMENTOS”, si se detecta la presencia de los dos elementos el sistema procederá a empernar la junta.

Al capturar las imágenes, el sistema convertirá la imagen de RGB a HSV ya que este código de colores facilita el análisis de las imágenes en el sistema, se filtra el color de los elementos en un rango de mínimos y máximos del código del color y se crea una máscara. Para determinar la existencia de los elementos se calcula un área, si dicha área es constante durante un determinado rango de tiempo el programa determina la existencia de elementos y se envía un dato de confirmación.

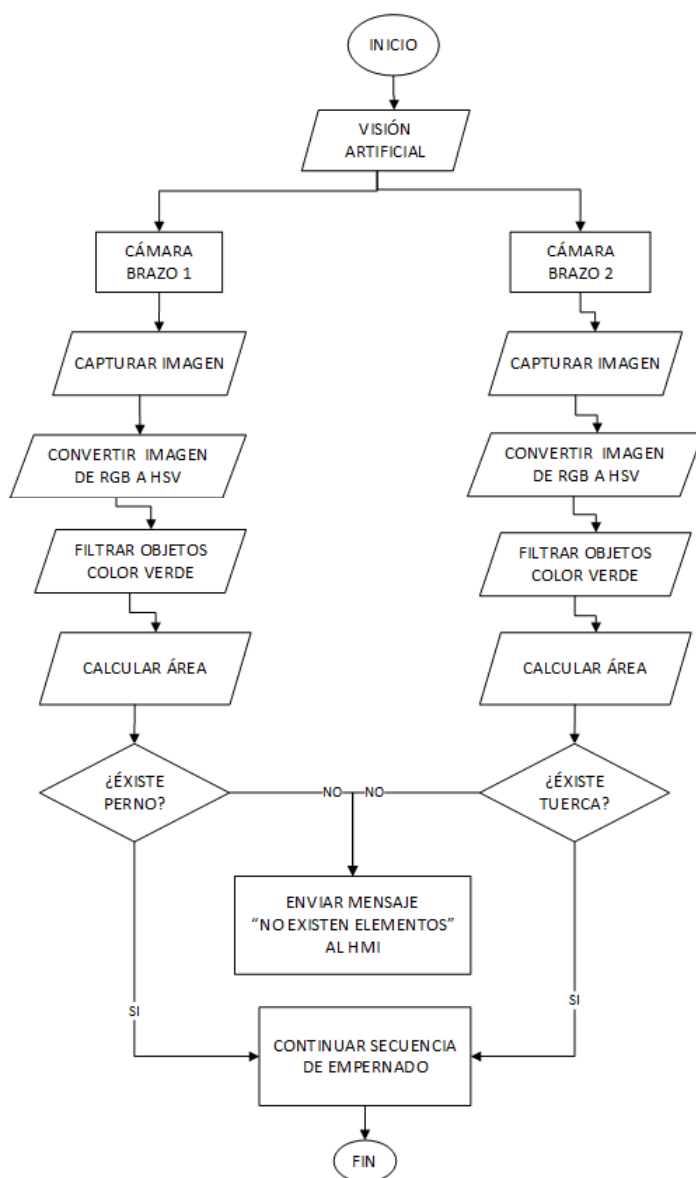


Figura 67: Diagrama de flujo, Visión artificial

3.5. Módulo 4. Planificador

3.5.1. Subsistema 4.1. Interfaz Gráfica

El HMI fue desarrollado con el uso del paquete de Python TkInter. Al iniciar el sistema se despliega la interfaz gráfica como se muestra en la Figura 68. En la interfaz gráfica se identifican dos tipos de operación del torso robótico, modo manual y modo automático.



Figura 68: HMI

3.5.1.1. Modo Manual

Como se muestra en la Figura 69 en el modo manual el usuario debe ingresar los datos de posición de cada brazo y luego presionar el botón “INGRESAR”. Si el usuario no ingresa los datos completos de los dos brazos, aparecerá un mensaje comunicando al usuario que debe “Ingresar datos completos”. Una vez calculada la cinemática inversa por medio de algoritmos genéticos, los datos de posición calculados se mostrarán en el HMI y el usuario podrá presionar el botón “EJECUTAR” para ubicar el torso robótico en la posición requerida.



Figura 69: Modo Manual

3.5.1.2. Modo Automático

Como se muestra en la Figura 70, el usuario tiene dos opciones de manejo automático del torso robótico Home y Secuencia. Al presionar el botón “HOME” el torso robótico ubicara los brazos en la posición inicial para empezar a trabajar. Al presionar el botón “SECUENCIA” el torso robótico iniciara una secuencia programada para empernar una junta.



Figura 70: Modo Automático

3.5.1.3. Botón Parar

El botón “PARAR” que se muestra en la Figura 71, se encuentra programado para parar el sistema en cualquier proceso que se este ejecutando.



Figura 71: Botón Parar

3.5.1.4. Mensajes de Información

Se crearon mensajes para informar al usuario sobre la inicialización y finalización de cada proceso. En el modo manual una vez ingresados los datos: a) el sistema inicializará el cálculo de cinemática inversa e indicará al usuario que el cálculo se ha inicializado mediante un mensaje de información, b) una vez terminado el cálculo el sistema informara al usuario que el proceso de cálculo ha terminado e indicará que debe presionar el botón “Ejecutar” para mover los motores a la posición deseada y c) una vez ubicados los motores se informará al usuario que la posición ha sido alcanzada Figura 72.

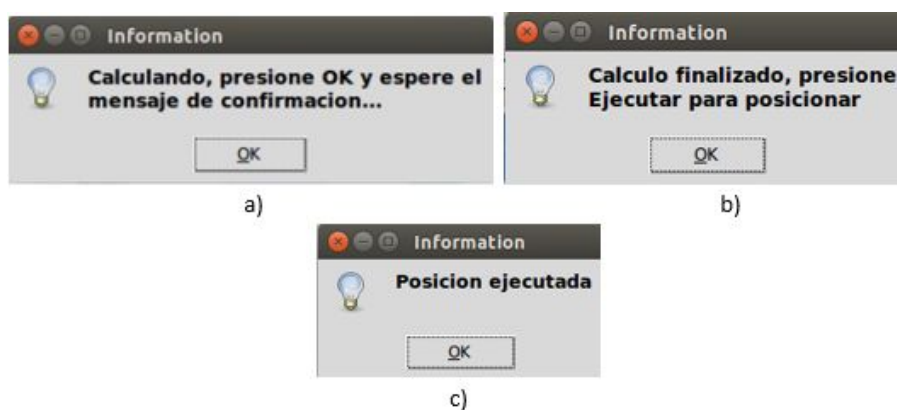


Figura 72: Mensaje de información: a) calculo inicializado, b) calculo terminado, c) posición ejecutada

En modo automático se crearon tres mensajes de información: a) mensaje de información una vez que el torso se ha ubicado en posición home, b) mensaje de información una vez que se ha inicializado la secuencia del proceso de empernaje y c) mensaje de información una vez que la secuencia a finalizado Figura 73.

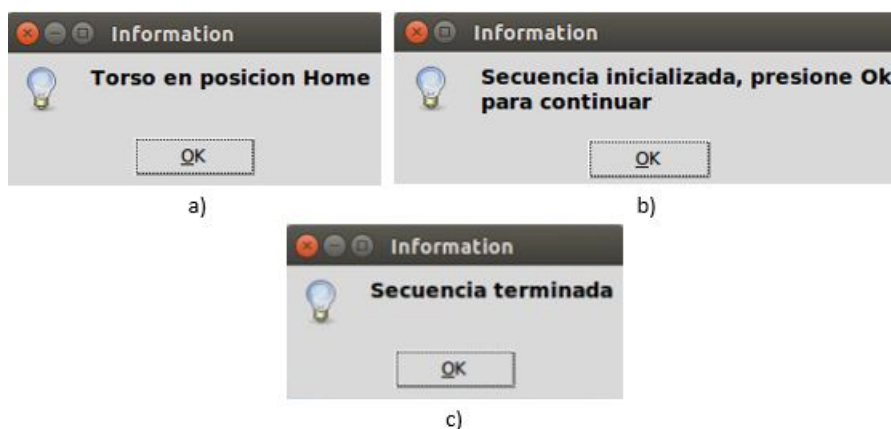


Figura 73: Mensaje de información: a) torso en posición Home, b) secuencia inicializada, c) secuencia terminada

3.5.1.5. Mensajes de información de estado del proceso

Se crearon mensajes para informar al usuario sobre el proceso de ubicación de los motores en cada estado, mientras los motores se posicionan aparecerá un mensaje en la parte inferior del HMI indicando que los motores se están "Ubicando", y una vez que los mismos se han posicionado el mensaje indicará que los motores se han "Ubicado" como se muestra en la Figura 74.

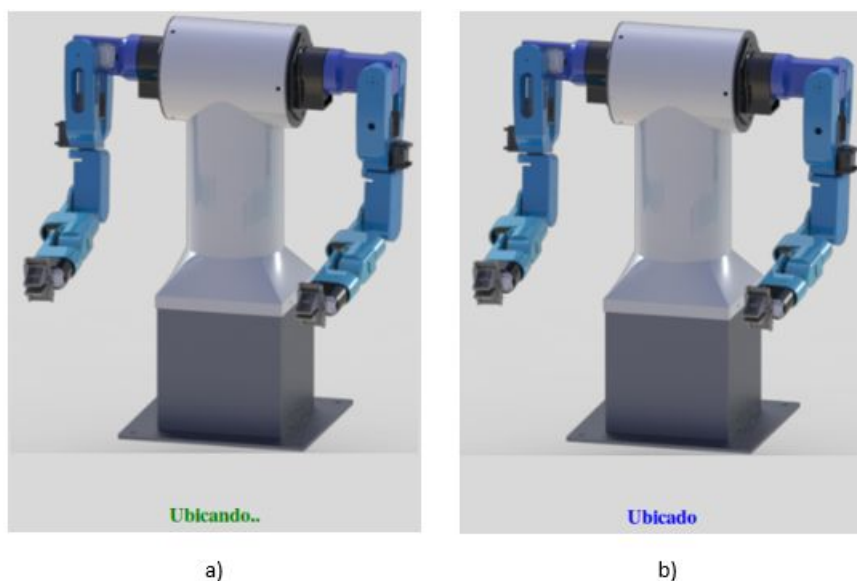


Figura 74: Mensajes de información del estado del proceso: a) Ubicando, b) Ubicado

3.5.1.6. Mensajes de Alerta

Los mensajes de alerta de la Figura 75 se visualizarán al correr la secuencia de modo automático, cuando el sistema de detección de elementos por medio de visión artificial, no encuentre los elementos necesarios para empernar la junta.



Figura 75: Mensajes de alerta

3.5.2. Subsistema 4.2. Máquina de Estados

El diagrama de flujo de la Figura 76 muestra la programación del sistema. El sistema tiene dos tipos de operación, Modo Manual y Modo Automático. En el modo manual el usuario debe ingresar datos de posición de cada brazo y pulsar el botón “Ingresar”, el sistema determinará si los datos ingresados se encuentran dentro del rango de trabajo y resolverá la cinemática por medio de algoritmos genéticos. Una vez realizado el cálculo de cinemática inversa el sistema mostrará en la interfaz gráfica los datos calculados e indicará al usuario que debe presionar el botón “Ejecutar” para posicionar los brazos, una vez pulsado dicho botón el sistema ubicará los brazos en la posición especificada. Si el usuario no ingresa los datos completos el sistema le indicará que

debe “Ingresar datos completos”.

En el modo automático el usuario tiene dos opciones Home y Secuencia, al pulsar “Home” los brazos se ubicaran en la posición inicial, determinada offline por medio de algoritmos genéticos, para empezar a trabajar. Al pulsar “Secuencia” los brazos seguirán una primera secuencia de posiciones offline determinadas por medio de algoritmos genéticos para posicionar las cámaras ubicadas en los efectores finales sobre los elementos perno y tuerca ubicados en la celda de trabajo. Una vez concluida la primera secuencia se activa el sistema de detección de elementos por medio de visión artificial. Si el sistema de detección de elementos detecta perno y tuerca enviará mensajes de confirmación al nodo secuencia para la activación de una segunda secuencia y continuar con el proceso de empernado. En caso de que el sistema de detección de elementos no detecte perno o tuerca activara mensajes de alerta, recordando al usuario colocar los elementos para continuar con la secuencia.

Si el usuario detecta visualmente algún problema en el funcionamiento o quiere parar el proceso, puede pulsar el botón “Parar” para detener el sistema en cualquier proceso que se esté ejecutando.

Los diferentes procesos del sistema se configuran como nodos ejecutables de ROS (Robot Operating System) para comunicarse con otros procesos mediante el uso de tópicos. Los tópicos son los canales de comunicación por medio de los cuales se transmiten los datos entre nodos. La conexión entre nodos y tópicos del sistema se detalla en el Apéndice C.

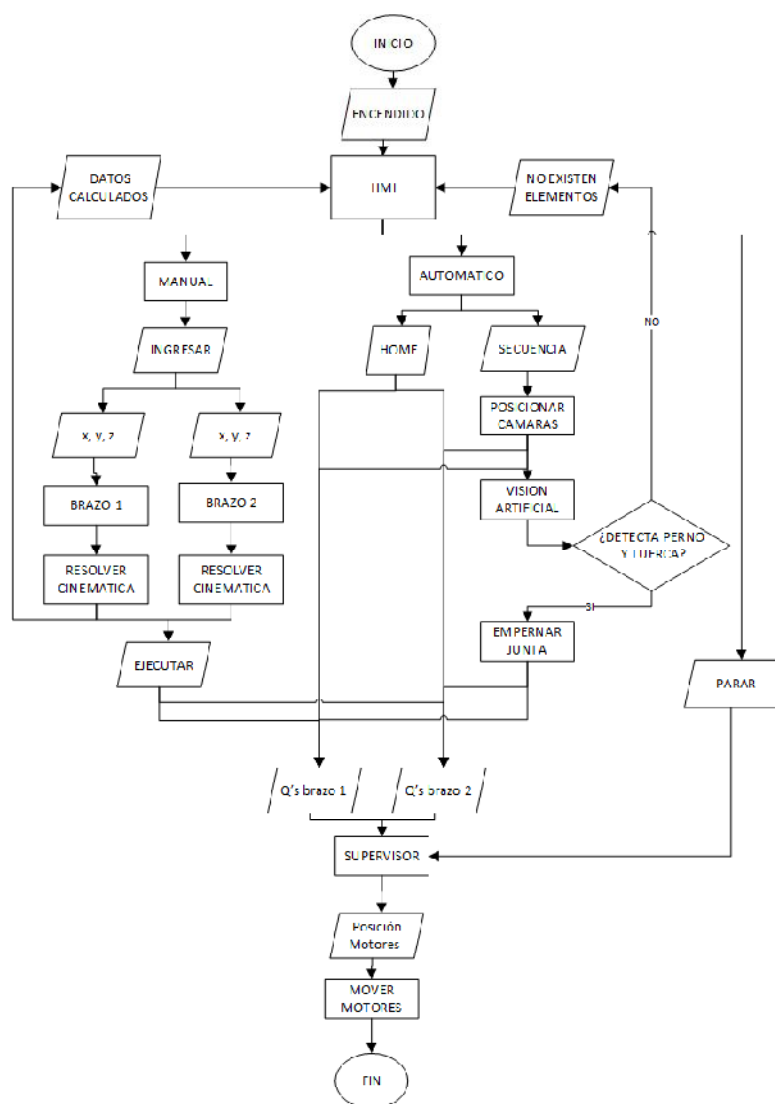


Figura 76: Diagrama de flujo del sistema

3.6. Resumen

En el capítulo se desarrolla el análisis de la metodología seleccionada para el desarrollo del diseño mecatrónico aplicado al presente proyecto. Se describe el desarrollo del proyecto mediante la metodología en V implementada, haciendo uso de las corrientes de especificación, desarrollo y pruebas aplicadas al presente estudio. Se presenta el despliegue de la función de calidad con la matriz resultante del estudio de las especificaciones y requerimientos del proyecto. Se presenta una estructura modular del proyecto que lo divide en subsistemas y detalla cada uno de ellos. Se describe el diseño de los módulos definidos como estructura, cinemática del manipulador, visión artificial y planificador.

Capítulo 4

Implementación, Pruebas y Resultados

El presente capítulo detalla el proceso de fabricación e implementación de los componentes mecánicos, eléctricos, electrónicos de cada módulo del torso robótico. Se realizan pruebas y se describe los resultados obtenidos en el presente proyecto.

4.1. Implementación

Se desarrolla la implementación del torso robótico de manera modular iniciando por la implementación de la estructura, mediante impresión 3D de los eslabones del segundo brazo y manufactura de la estructura mecánica, la misma que se detalla en hojas de procesos y cuyos planos de construcción se detallan en anexos. Se implementa los componentes electrónicos del torso y se realiza pruebas de funcionamiento del sistema completo.

4.1.1. Estructura

Se realizó la implementación del segundo brazo antropomorfo mediante impresión 3D con los parámetros que se muestran en la Tabla 29. Los eslabones del brazo original realizado en el proyecto [Sánchez & Terán, 2015] fueron duplicados por medio de impresión en la máquina de prototipado rápido Prusa i3. En la Figura 76 se detalla la configuración de la impresora 3D y en la Figura 77 se observa la impresión en curso del eslabón dos del manipulador.

Tabla 29.
Parámetros de impresión

Parámetro	Especificación
Material	PLA
Espesor de capa	0.2mm
Densidad de impresión	80%

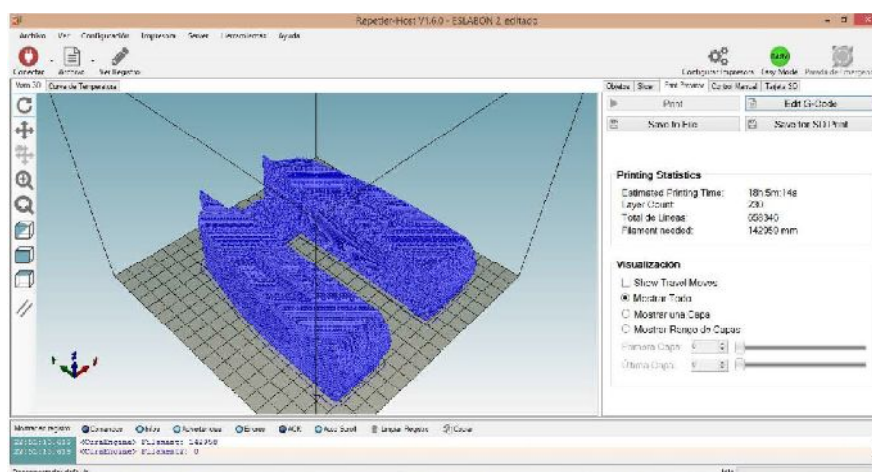


Figura 76: Configuración para la impresión 3D de los eslabones del segundo brazo del manipulador colaborativo

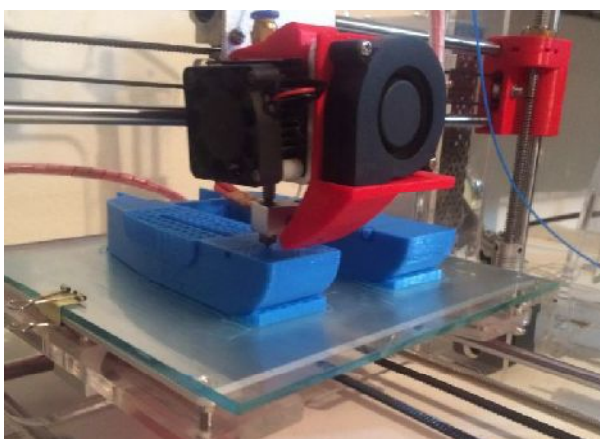


Figura 77: Impresión 3D de los eslabones del segundo brazo del manipulador colaborativo

Una vez terminada la impresión de todos los eslabones se procedió a realizar el ensamble de las piezas con los motores y su efector final de acuerdo a las especificaciones de construcción definida en el proyecto [Sánchez & Terán, 2015].

Determinado el diseño final del torso del manipulador colaborativo se procedió a realizar la construcción del mismo. El proceso de manufactura de las piezas implicó el uso de varias máquinas y herramientas del laboratorio de manufactura de la Universidad de las Fuerzas Armadas ESPE. Se usaron herramientas como regla ingenieril, escuadra, compás, martillo, granete, lijas y limas. Se realizaron procesos de corte por plasma, corte mediante moladora y corte por cizalla, proceso de doblado en la dobladora de metales, proceso de perforado en el taladro de banco y el taladro de mano, proceso de soldadura mediante soldadura TIG y soldadura SMAW, proceso de rolado mediante la roladora de metales, proceso de pulido mediante lijas, limas y la moladora con el uso del disco de pulido, proceso de masillado y proceso de pintado mediante soplete. En la Figura 78 se presenta un resumen de los diferentes procesos utilizados en la implementación de la estructura.

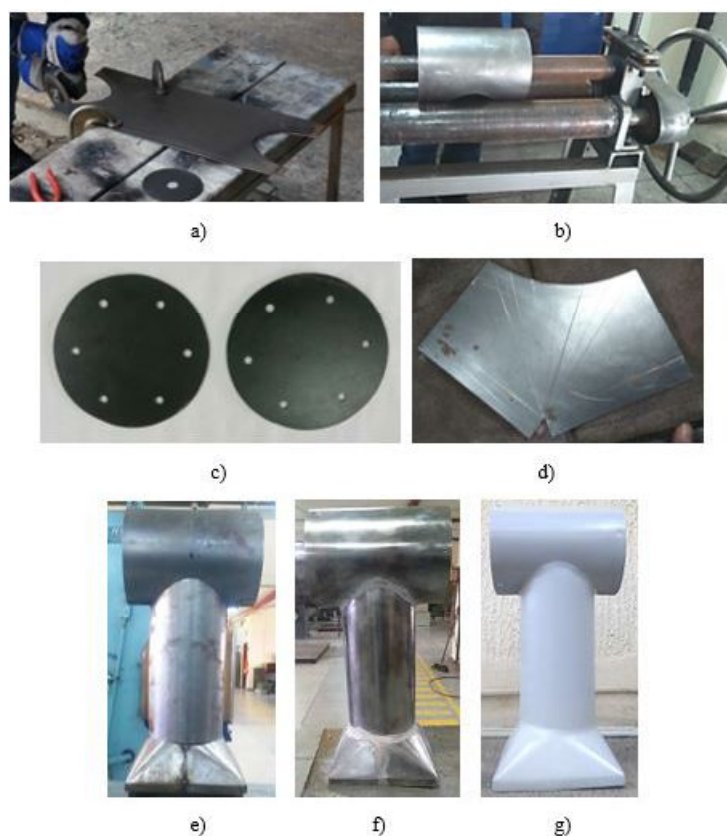


Figura 78: Procesos de a) corte b) rolado c) perforación d) trazado y doblado e) soldadura f) masillado g) pintura

El proceso de construcción se describe en las hojas de procesos en donde se detalla las dimensiones, el material, la secuencia de operaciones, las máquinas y herramientas utilizadas y el tiempo de manufactura de cada elemento. En la Tabla 30 se detalla el proceso de manufactura de la base del torso, en la Tabla 31 el proceso de manufactura

del torso, y en la Tabla 32 el proceso de manufactura de las tapas del torso.

Tabla 30.

Hoja de Procesos, Base del Torso



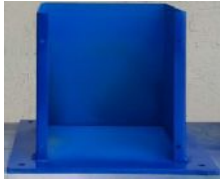
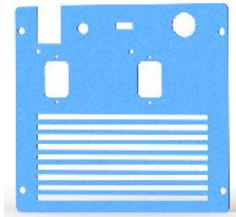
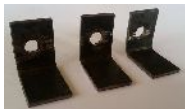

HOJA DE PROCESOS - BASE DEL TORSO				
Dimensiones	300 x 300 x 220 [mm]			
Encargados	Anahí Carvajal M. Sonia Martínez P.			
Elemento	Material	Secuencia de operaciones	Máquinas & Herramientas	Tiempo
Base estructura 	Acero ASTM A36 6 mm	Medir y marcar límites y puntos de perforación	Flexómetro Regla metálica Rayador Compás	30 min
		Cortar	Corte por plasma	120 min
		Perforar	Taladro de banco Brocas	60 min
		Limar Lijar	Lima plana Lijas	60 min
Base en C 	Acero ASTM A36 2 mm	Medir y marcar límites y puntos de perforación	Flexómetro Regla metálica Rayador Compás	30 min
		Cortar	Corte por plasma	120 min
		Doblar	Dobladora de metal	30 min
		Perforar	Taladro de banco Brocas	60 min
		Limar Lijar	Lima plana Lijas	60 min
Sub ensamble base 	Electrodos	Soldar	Soldadora SMAW	15 min
	Disco de pulir	Lijar	Lijas	30 min
	Pintura Thinner	Pulir	Moladora	30 min
Pintar		Soplete	60 min	
Panel de Control				
	Acrílico	Cortar	Cortadora láser	10 min

Tabla 31.
Hoja de Procesos, Torso

HOJA DE PROCESOS - TORSO				
Dimensiones	440 x 200 x 204 [mm]			
Encargados	Anahí Carvajal M. Sonia Martínez P.			
Elemento	Material	Secuencia de operaciones	Máquinas & Herramientas	Tiempo
Chapa transición 	Acero ASTM A36 1.2mm Disco de corte	Trazar planos	Regla metálica	60 min
		Medir y marcar límites y puntos	Rayador Compás Cizalla	
		Cortar	Moladora	120 min
		Limar	Lima curva	20 min
		Doblar	Dobladora de metal	120 min
Ensamble Chapa transición	Electrodos	Soldar	Soldadura TIG Pinzas de presión	60 min
				
Cilindros vertical y horizontal 	Acero ASTM A36 1.2 mm Disco de corte	Trazar planos	Regla metálica	180 min
		Medir y marcar límites y puntos	Rayador Compás	
		Cortar	Moladora	180 min
		Limar	Lima curva	60 min
		Rolar	Roladora de metal	120 min
Ensamble Cilindros	Electrodos	Soldar	Soldadura TIG Pinzas de presión	60 min
				
Sub ensamble Torso 	Electrodos Masilla Disco de pulir Pintura Thinner	Soldar	Soldadura TIG Pinzas de presión	60 min
		Perforar	Taladro de banco Brocas	30 min
		Masillar	Masilla Secante	120 min
		Lijar	Lijas	60 min
		Pulir	Moladora	30 min
		Pintar	Soplete	60 min

Tabla 32.
Hoja de Procesos, Tapas laterales del Torso

HOJA DE PROCESOS - TAPAS LATERALES DEL TORSO				
Dimensiones	∅ 160 x 24 [mm]			
Encargados	Anahí Carvajal M. Sonia Martínez P.			
Elemento	Material	Secuencia de operaciones	Máquinas & Herramientas	Tiempo
Tapas laterales 	Acero ASTM A36 2 mm Brocas	Trazar planos	Regla metálica	60 min
		Medir y marcar límites y puntos	Rayador Compás Cizalla	
		Cortar	Esmeril	240 min
		Limar	Lima curva	30 min
		Lijar	Lijas	30 min
		Perforar	Taladro de banco Brocas	60 min
Sujeciones 	Acero ASTM A36 2 mm	Trazar planos	Regla metálica	60 min
		Medir y marcar límites y puntos	Rayador Compás	
		Cortar	Cizalla	60 min
		Limar	Lima plana	30 min
		Lijar	Lijas	30 min
		Perforar	Taladro de banco Brocas	60 min
		Doblar	Dobladora de metal	60 min
Sub ensamble tapas laterales 	Electrodos	Soldar	Soldadura SMAW	60 min
	Disco de pulir		Pinzas de presión	
	Pintura	Pulir	Lijas	
Thinner		Moladora		30 min
		Pintar	Soplete	60 min

Posterior a los diferentes procesos realizados para obtener las piezas del torso, se procede a ensamblar las partes. Se unen los dos brazos antropomorfos mediante bridas a cada lado de la estructura para obtener como resultado el manipulador robótico colaborativo que se muestra en la Figura 79



Figura 79: Implementación del torso robótico colaborativo de dos brazos antropomorfos completo

4.1.2. Implementación Acople para Cámara

La implementación del diseño del gripper se realizó mediante impresión 3D en una impresora delta MakerBot, el material utilizado para la impresión fue PLA de color negro. Una vez impreso el acople diseñado se procedió a colocar la cámara, el resultado se observa en la Figura 80. La cámara ingresa en el acople correctamente, con una sujeción suficiente para posicionar el dispositivo en cualquier posición sin que se caiga o sufra algún daño. El acople diseñado e implementado para adaptar la cámara de visión artificial cumple con la estabilidad necesaria para desarrollar el proceso de adquisición de imágenes.



Figura 80: Acople cámara implementación

4.1.3. Panel de Control

El panel de control del torso robótico está constituido como se muestra en la Figura 81. Se encuentra en la parte posterior del torso y fue implementado con acrílico transparente de color azul para identificar los componentes internos. El diseño del mismo fue desarrollado para ser de fácil montaje, por lo que se usaron pernos hexagonales Allen M5x16.

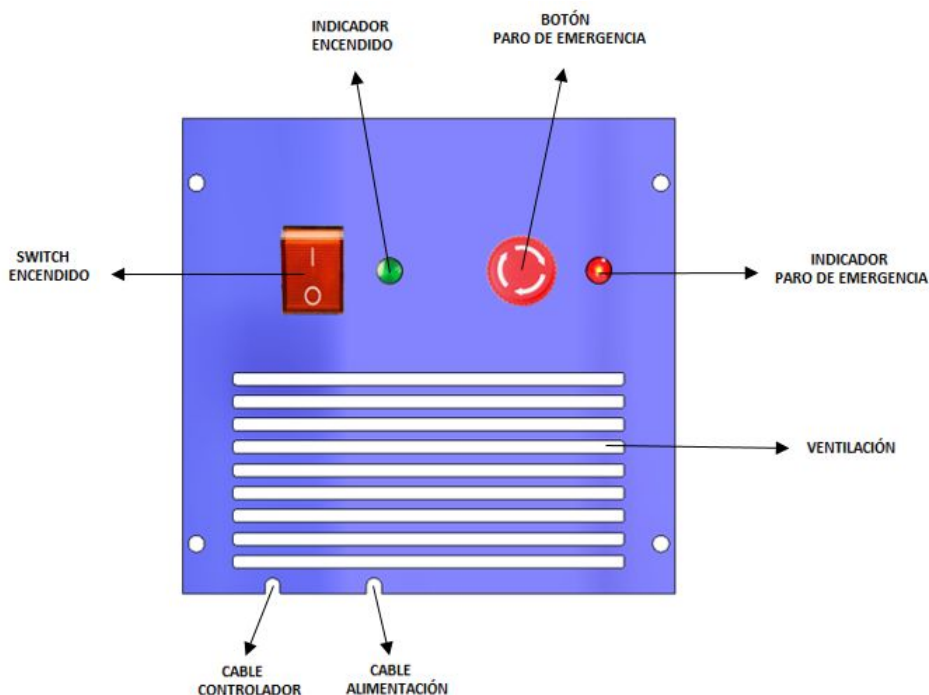


Figura 81: Panel de Control

El panel de control está constituido por un switch de encendido (SWITCH ENCENDIDO) con el cual se alimenta todo el sistema. Posee un indicador de encendido (INDICADOR ENCENDIDO) color verde para indicar cuando el sistema compuesto por los dos brazos antropomorfos y el controlador estén alimentados. Como elementos de seguridad se tiene un botón (BOTÓN PARO DE EMERGENCIA) y un indicador (INDICADOR PARO DE EMERGENCIA) de paro de emergencia, ambos de color rojo. El botón de paro de emergencia bloquea la alimentación del sistema al ser presionado e inmediatamente se activa el indicador de paro de emergencia y se envía un mensaje de error a la interfaz gráfica. En la parte inferior se encuentran los agujeros dispuestos para la ventilación (VENTILACIÓN), el cable del controlador USB2Dynamixel (CABLE CONTROLADOR) y el cable de alimentación a 110VAC (CABLE ALIMENTACIÓN).

4.1.4. Placa PCB

EL diseño de la placa fue realizado en función de los requerimientos de los componentes electrónicos a utilizar. La placa fue implementada en fibra y etiquetada para su correcta conexión como se muestra en la Figura 82.



Figura 82: Placa del circuito electrónico

EL diseño fue desarrollado e impreso en función de las especificaciones del circuito. La Placa contiene cuatro borneras asignadas para comunicación y alimentación del sistema compuesto por los dos brazos antropomorfos y el controlador de los motores. La primera bornera de dos entradas del lado izquierdo está asignada al controlador (CT), contiene entrada de datos (DATA) y GND. La segunda bornera corresponde al brazo uno con sus tres pines GND, VDD, DATA y de la misma forma la tercera bornera para el brazo dos en orden DATA, VDD, GND, de izquierda a derecha. La última bornera está asignada a la alimentación del sistema dada por la fuente de 12V y 6A que consta de dos entradas VDD y GND.

4.2. Pruebas y Resultados

Una vez implementado el torso robótico, se evalúa su funcionamiento realizando varias pruebas:

1. Primera prueba, funcionamiento de los componentes instalados en el sistema.
2. Segunda prueba, funcionamiento del programa de algoritmos genéticos para la resolución de la cinemática inversa de los brazos.
3. Tercera prueba, sistema de detección de objetos por medio de visión artificial.
4. Cuarta prueba, comunicación entre nodos del sistema.

5. Quinta prueba, funcionamiento general del sistema.

4.2.1. Primera prueba, funcionamiento de los componentes instalados en el sistema

La primera prueba se realiza para comprobar el funcionamiento de los componentes instalados en el sistema. Luego de haber instalado los motores, controlador USB2Dynamixel, fuente de alimentación, y panel de control, se evalúa el funcionamiento de los mismos como conjunto. Se comprueba la correcta conexión de los componentes del panel de control. Se comprueba además la correcta asignación de IDs a cada motor, la cantidad de motores Dynamixel AX-12, MX-28 y MX-64.

- Para probar la correcta conexión de los componentes instalados, se realizó una medición de corriente y voltaje de salida de la fuente, de entrada a los motores, el voltaje de comunicación del controlador, voltaje para comprobar el funcionamiento del paro de emergencia.

Tabla 33.
Corriente y Voltaje de los componentes instalados

Elemento	Voltaje	Corriente
Fuente de alimentación	12 VDC	6 A
Motores	12 VDC	Varia dependiendo del torque
Controlador	5 VDC	
Paro de emergencia pulsado	0 VAC	
Paro de emergencia sin pulsar	110 VAC	

Como se observa en la Tabla 33, la fuente de alimentación es de 12 VDC. Los actuadores se alimentan de 12 VDC y su corriente varía dependiendo del torque que están realizando los motores. El cable de datos del controlador marca 5 VDC, voltaje necesario para enviar datos a los motores y controlar su posicionamiento. Al pulsar el paro de emergencia marca 0 VAC, lo cual comprueba su correcto funcionamiento, desenergizando temporalmente el sistema.

- Para comprobar la conexión de los motores, y la correcta asignación de IDs se ejecutó un programa del paquete de dynamixel llamado `motores_conectados.launch` que detecta los IDs de los motores y su modelo.

```

salome@salome-RC512:~$ roslaunch motores motores_conectados.launch
... Logging to /home/salome/.ros/log/9e579f72-93e5-11e6-8f5b-64809913264c/roslaunch-salome-RC512-9614.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://salome-RC512:36108/

SUMMARY
-----
PARAMETERS
 * /dynamixel_manager/namespace: dxl_manager
 * /dynamixel_manager/serial_ports/pan_tilt_port/ baud_rate: 1000000
 * /dynamixel_manager/serial_ports/pan_tilt_port/max_motor_id: 25
 * /dynamixel_manager/serial_ports/pan_tilt_port/min_motor_id: 1
 * /dynamixel_manager/serial_ports/pan_tilt_port/port_name: /dev/ttyUSB0
 * /dynamixel_manager/serial_ports/pan_tilt_port/update_rate: 20
 * /roscpplog: jade
 * /rosversion: 1.11.20

NODES
 /
  dynamixel_manager (dynamixel_controllers/controller_manager.py)

auto-starting new master
process[master]: started with pid [9625]
ROS_MASTER_URI=http://localhost:11311

setting /run_id to 9e579f72-93e5-11e6-8f5b-64809913264c
process[rosout-1]: started with pid [9638]
started core service [/rosout]
process[dynamixel_manager-2]: started with pid [9641]
[INFO] [WallTime: 1476652512.493515] pan_tilt_port: Pinging motor IDs 1 through 25...
[INFO] [WallTime: 1476652514.130124] pan_tilt_port: Found 16 motors - 8 AX-12 [5, 6, 7, 8, 15, 16, 17, 18], 2 MX-28 [4, 14], 6 MX-64 [1, 2, 3, 11, 12, 13], initialization complete.

```

Figura 83: Conexión de motores desde el terminal

Como se muestra en la Figura 83, al estar conectados los dos brazos se encuentran 16 motores: 8 AX-12, 2 MX-28 y 6 MX-64.

4.2.2. Segunda prueba, funcionamiento del programa de algoritmos genéticos para la resolución de la cinemática inversa de los brazos

La segunda prueba se realiza para comprobar la correcta sintonización del programa desarrollado para la resolución de cinemática inversa de los brazos por medio de algoritmos genéticos.

Para la implementación del algoritmo se modificó el máximo número de iteraciones, el porcentaje de mutación y el rango de creación de individuos de la población inicial. El algoritmo utilizado converge a las 200 iteraciones con un tiempo aproximado de un segundo y medio. Inicialmente se codificó el algoritmo genético sin limitaciones, es decir que el rango asignado para cada actuador fue de 0° a 360° . Sin embargo, al momento de implementar el posicionamiento físico del manipulador fue necesario incorporar los límites de posición angular para cada motor en función del volumen de trabajo del torso. Los límites determinados para los motores del primero, segundo y tercer eslabón fueron: $0^\circ - 40^\circ$, $0^\circ - 20^\circ$, $0^\circ - 180^\circ$, respectivamente. Mientras a los demás motores fueron asignados rangos entre $0^\circ - 200^\circ$. En la Figura 84 se observa el resultado de la resolución de la cinemática inversa de un brazo posicionado en un punto determinado como inaccesible físicamente. El error cuadrático medio, expresado como “fitness”, generado en el primer caso ($0^\circ - 360^\circ$) es de 0.004631, mientras el error generado en el caso real es de 31.226777.

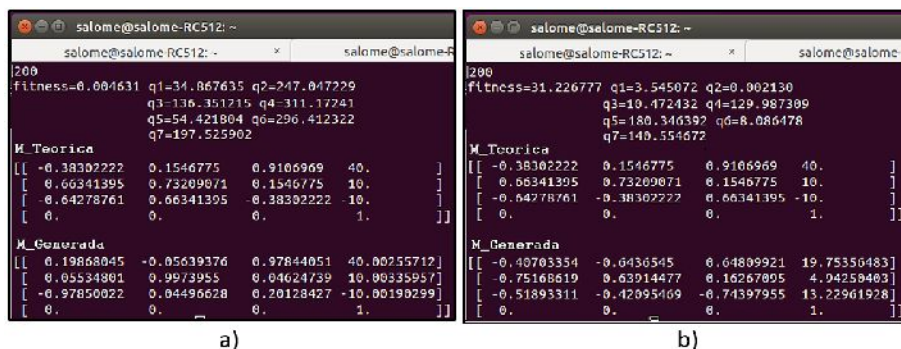


Figura 84: Resolución de cinemática inversa mediante algoritmos genéticos – posición inaccesible a) algoritmo sin límites b) algoritmo con límites (caso real)

Una vez identificados los ejes de orientación y el volumen de trabajo real aproximado, se definieron las posiciones a las cuales puede acceder el manipulador físicamente. Después del análisis desarrollado se determina que el algoritmo genético implementado con límites angulares para el caso real, converge a la solución de mejor forma que el algoritmo ideal. Como se puede observar en la Figura 85, el error cuadrático medio resultante del posicionamiento del manipulador, expresado como “fitness”, es de 0.003337 y el error cuadrático medio en el caso real es de 0.000721. Ambos algoritmos convergen a la solución del posicionamiento del manipulador, tanto en el caso ideal como en el caso real. Sin embargo el algoritmo limitado converge en menor tiempo y con mayor precisión.

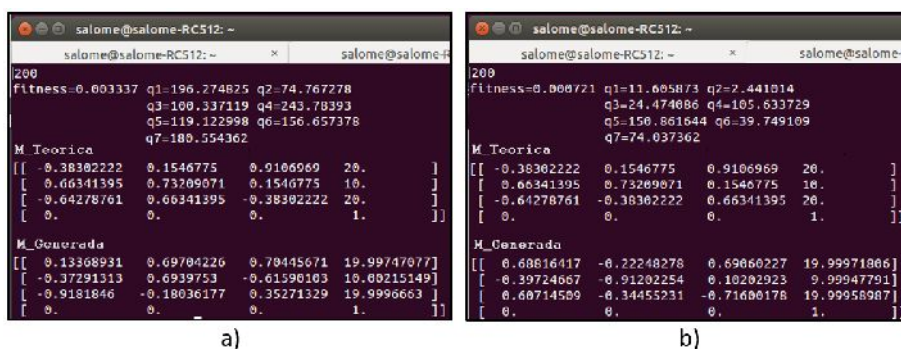


Figura 85: Resolución de cinemática inversa mediante algoritmos genéticos – posición accesible a) algoritmo sin límites b) algoritmo con límites (caso real)

4.2.3. Tercera prueba, sistema de detección de objetos por medio de visión artificial

La tercera prueba se realiza para comprobar la correcta sintonización del color de perno y tuerca en el programa desarrollado para detección de objetos por medio de visión artificial. Se desarrolló un programa de detección de colores en código HSV

(Hue, Saturation, Valor) para filtrar los objetos, se maneja en un rango de altos y bajos. Al tener elementos de color verde se inicia el filtrado con los rangos que se muestra en la Tabla 34, y se obtiene la máscara de la imagen filtrada que se muestra en la Figura 86.

Tabla 34.
Rangos iniciales de color verde

Rango	H	S	V
Bajos	49	50	50
altos	80	255	255



Figura 86: Resultado de filtro con rangos iniciales

Después de modificar los valores de los rangos de color, se llegó a los valores correctos de color verde, los mismos que se muestran en la Tabla 35, y se obtiene la máscara de la imagen filtrada que se muestra en la Figura 87.

Tabla 35.
Rangos finales de color verde

Rango	H	S	V
Bajos	50	50	50
altos	102	200	180

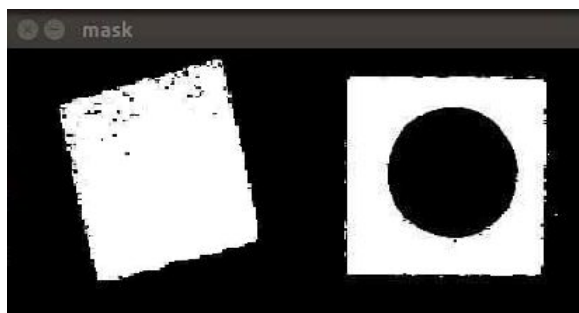


Figura 87: Resultado de filtro con rangos finales

4.2.4. Cuarta prueba, comunicación entre nodos del sistema

La cuarta prueba se realiza para comprobar la conexión entre nodos y tópicos del sistema mediante el rqt de ROS, rqt es un framework que permite visualizar la conexión en una forma semejante a un diagrama de flujo.

Los diferentes procesos del sistema se configuraron como nodos ejecutables de ROS (Robot Operating System), enviando datos mediante tópicos entre ellos. Para el sistema Torso Robótico se utilizaron tópicos tipo Pose, PoseArray, Bool y Goals . La Figura 88 muestra los tópicos que se usaron en el sistema, mientras en la Figura 89 muestra la conexión que se está realizando entre nodos y la correcta comunicación entre ellos.



Figura 88: Tópicos del sistema Torso Robótico

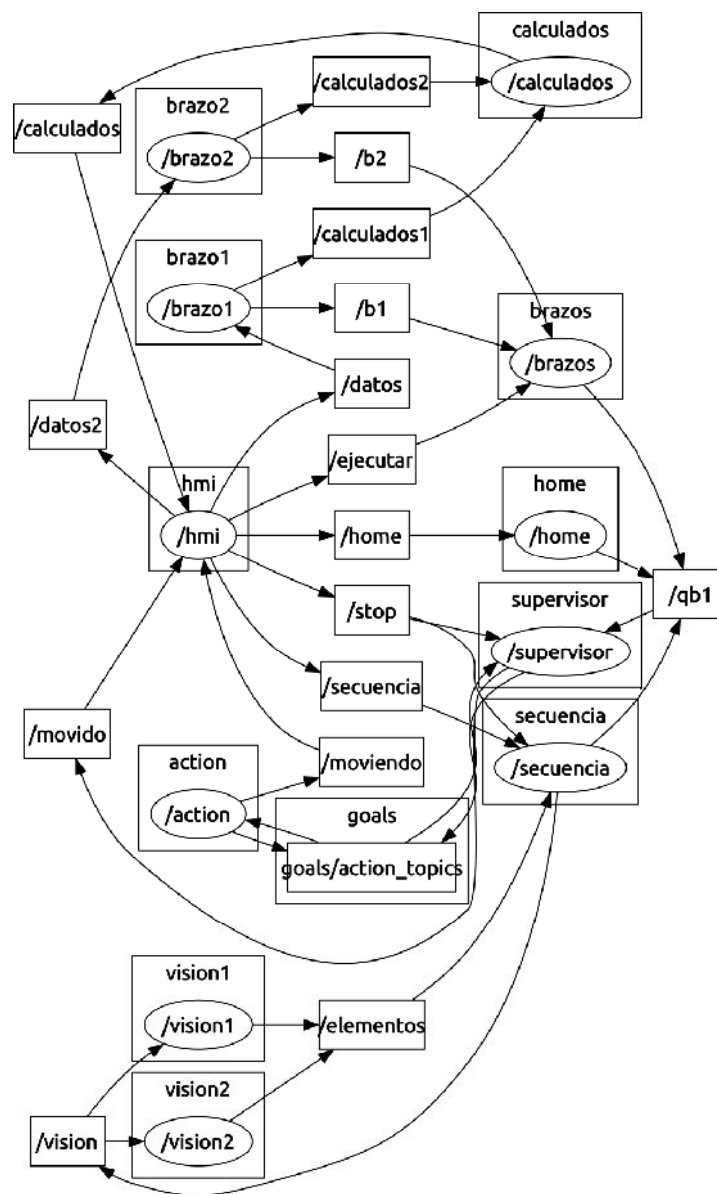


Figura 89: Visualización de la comunicación entre nodos del sistema Torso Robótico

Los nodos son programas ejecutables que realizan funciones concretas, dichos nodos se enlazan entre ellos por medio de tópicos para envío y recepción de mensajes de manera modular y son compilados, ejecutados y gestionados por un nodo principal ROS Master. Los nodos implementados en el sistema del torso robótico se detallan a continuación en la Tabla 36.

Tabla 36.
Nodos ROS del torso robótico

Nodo	Descripción
/hmi	Inicializa el sistema e inicializa la interfaz gráfica
/brazo1	Recibe datos de posición y orientación del brazo 1 de modo manual y resuelve la cinemática inversa del mismo por medio de algoritmos genéticos
/brazo2	Recibe datos de posición y orientación del brazo 2 de modo manual y resuelve la cinemática inversa del mismo por medio de algoritmos genéticos
/brazos	Recibe datos de la resolución de la cinemática inversa de los brazos y los envía al supervisor para su ejecución
/calculados	Recibe los datos de posicionamiento calculados y los envía al hmi para que el usuario pueda visualizarlos
/supervisor	Recibe las Qs del sistema y los manda al nodo action
/action	Recibe los datos de supervisor, inicializa los motores y los posiciona
/secuencia	Realiza una secuencia para empernar una junta e inicializa los nodos de visión
/vision1	Detecta presencia de perno en la junta
/vision2	Detecta presencia de tuerca en la junta
/home	Envía datos de posición inicial al supervisor

4.2.5. Quinta prueba, funcionamiento general del sistema

La quinta prueba se realiza para comprobar el funcionamiento general del sistema. El sistema tiene dos tipos de operación modo manual y modo automático. En el HMI de la Figura 90, se muestran cinco botones:

1. Ingresar: Ingresa datos de posición de cada brazo para realizar el cálculo de cinemática inversa.
2. Ejecutar: Ubica los brazos en la posición ingresada por el usuario
3. Home: Posiciona el Torso en la posición home.
4. Secuencia: Ubica las cámaras de los efectores finales sobre perno y tuerca, detecta la existencia de estos elementos y continua con el proceso de empernado.
5. Parar: Detiene el sistema en cualquier proceso que se esté ejecutando.



Figura 90: HMI Torso Robótico

Para el funcionamiento en modo manual se realizó la programación para que el usuario ingrese datos de posición de cada brazo y presione el botón “INGRESAR”. Una vez ingresados los datos el sistema inicializa el cálculo de cinemática inversa de cada brazo por medio de algoritmos genéticos e informa al usuario que el cálculo se ha inicializado Figura 91, una vez que el cálculo ha terminado se visualizarán los datos calculados en el HMI y se informará al usuario que el cálculo ha terminado y que puede presionar el botón “EJECUTAR” para ubicar los motores Figura 92, una vez ubicados los motores se informa al usuario que el torso se ha ubicado en la posición requerida Figura 93.



Figura 91: Información, inicio de cálculo de cinemática



Figura 92: Información, cálculo finalizado



Figura 93: Información, posición ejecutada

Para comunicar al usuario sobre el proceso de ubicación de los motores en cada estado, se mostrará un mensaje en la parte inferior del HMI indicando que los motores se están “Ubicando” Figura 94, y una vez que los mismos se han posicionado el mensaje indicará que los motores se han “Ubicado” Figura 95. Además para comprobar que los datos de posicionamiento están llegando al nodo action, se imprimen los mismos en el terminal como se muestra en la Figura 96.



Figura 94: Información de posicionamiento, ubicando



Figura 95: Información de posicionamiento, ubicado

```
[INFO] [wallTime: 1476661047.199292] Moviendo brazos
Llevo
motor1: 142 motor2: 149 motor3: 178 motor4: 205 motor5: 155 motor6: 280 motor7: 239 gripper1: 400
motor11: 172 motor12: 166 motor13: 156 motor14: 231 motor15: 140 motor16: 273 motor17: 162 gripper2: 400
[INFO] [wallTime: 1476661047.200057] Succeeded
cliente hecho
```

Figura 96: Posicionamiento de cada estado

El funcionamiento de home consiste en un nodo que guarda las posiciones iniciales del sistema determinadas offline por medio de algoritmos genéticos. Al aplastar home el nodo activa el envío de dichas posiciones al nodo supervisor, este nodo manda a mover los motores en el nodo action e informa al usuario que el torso ha sido ubicado en esta posición Figura 97. Los datos que recibe el nodo action se muestran en la Figura 98.



Figura 97: Información, torso en Home

```
[INFO] [WallTime: 1476661122.249936] Moviendo brazos
llego
motor1: 286 motor2: 1041 motor3: 1123 motor4: 1344 motor5: 583 motor6: 524 motor7: 528 gripper1: 400
motor11: 578 motor12: 3968 motor13: 2684 motor14: 635 motor15: 512 motor16: 520 motor17: 817 gripper2: 468
[INFO] [WallTime: 1476661122.251047] Succeeded
cliente hecho
```

Figura 98: Funcionamiento Home

Al apretar Secuencia el sistema informa al usuario que el proceso ha sido inicializado Figura 99, y ubica las cámaras sobre los elementos en una secuencia de posiciones offline determinadas por medio de algoritmos genéticos, el seguimiento de la secuencia se observa en la Figura 100.



Figura 99: Información, secuencia inicializada

```

secuencia
0
[INFO] [wallTime: 1476719563.776777] Moviendo brazos
llego
motor1: 286 motor2: 1041 motor3: 1123 motor4: 1341 motor5: 500 motor6: 524 motor7: 528 gripper1: 400
motor11: 570 motor12: 3908 motor13: 2884 motor14: 635 motor15: 508 motor16: 520 motor17: 817 gripper2: 400
[INFO] [wallTime: 1476719563.778005] Succeeded
cliente hecho
1
[INFO] [wallTime: 1476719569.798746] Moviendo brazos
llego
motor1: 286 motor2: 1041 motor3: 79 motor4: 1341 motor5: 500 motor6: 524 motor7: 528 gripper1: 400
motor11: 570 motor12: 3908 motor13: 3870 motor14: 635 motor15: 508 motor16: 520 motor17: 817 gripper2: 400
[INFO] [wallTime: 1476719569.799371] Succeeded
cliente hecho
2
[INFO] [wallTime: 1476719575.782465] Moviendo brazos
llego
motor1: 286 motor2: 1041 motor3: 79 motor4: 254 motor5: 500 motor6: 829 motor7: 528 gripper1: 400
motor11: 570 motor12: 3908 motor13: 3870 motor14: 1706 motor15: 508 motor16: 200 motor17: 817 gripper2: 400
[INFO] [wallTime: 1476719575.783555] Succeeded
cliente hecho
3
[INFO] [wallTime: 1476719581.803571] Moviendo brazos
llego
motor1: 286 motor2: 1041 motor3: 1140 motor4: 254 motor5: 500 motor6: 829 motor7: 528 gripper1: 400
motor11: 570 motor12: 3908 motor13: 2900 motor14: 1706 motor15: 508 motor16: 200 motor17: 817 gripper2: 400
[INFO] [wallTime: 1476719581.804844] Succeeded
cliente hecho
4
[INFO] [wallTime: 1476719587.788341] Moviendo brazos
llego
motor1: 286 motor2: 1041 motor3: 1140 motor4: 254 motor5: 500 motor6: 829 motor7: 822 gripper1: 400
motor11: 570 motor12: 3908 motor13: 2900 motor14: 1706 motor15: 508 motor16: 200 motor17: 514 gripper2: 400
[INFO] [wallTime: 1476719587.789657] Succeeded
data: true
vision inicializada
vision brazo uno inicializado
vision brazo dos inicializado
tuerca disponible
hay tuerca
perno disponible
hay perno
2
hay elementos
secuencia
0
[INFO] [wallTime: 1476719619.996943] Moviendo brazos
llego
motor1: 65 motor2: 1025 motor3: 1140 motor4: 700 motor5: 500 motor6: 760 motor7: 822 gripper1: 400
motor11: 715 motor12: 3973 motor13: 2929 motor14: 1356 motor15: 508 motor16: 205 motor17: 514 gripper2: 400
[INFO] [wallTime: 1476719619.997842] Succeeded
cliente hecho

```

Figura 100: Funcionamiento secuencia primera parte

Una vez terminada la primera secuencia se inicializan los nodos de visión artificial para detectar la presencia de elementos, si hay elementos el sistema envía datos de confirmación para inicializar la segunda parte de la secuencia, esta activación se comprueba en la Figura 106. En caso de que el sistema de detección de elementos no encuentre los mismos enviará mensajes de alerta al usuario indicándole que debe colocar el perno Figura 102 o que debe colocar la tuerca Figura 103 para continuar con el proceso.

```

4
[INFO] [wallTime: 1476719587.788341] Moviendo brazos
llego
motor1: 286 motor2: 1041 motor3: 1140 motor4: 254 motor5: 500 motor6: 829 motor7: 822 gripper1: 400
motor11: 570 motor12: 3908 motor13: 2900 motor14: 1706 motor15: 508 motor16: 200 motor17: 514 gripper2: 400
[INFO] [wallTime: 1476719587.789657] Succeeded
cliente hecho
data: true
vision inicializada
vision brazo uno inicializado
vision brazo dos inicializado
tuerca disponible
hay tuerca
perno disponible
hay perno
2
hay elementos
secuencia
0
[INFO] [wallTime: 1476719619.996943] Moviendo brazos
llego
motor1: 65 motor2: 1025 motor3: 1140 motor4: 700 motor5: 500 motor6: 760 motor7: 822 gripper1: 400
motor11: 715 motor12: 3973 motor13: 2929 motor14: 1356 motor15: 508 motor16: 205 motor17: 514 gripper2: 400
[INFO] [wallTime: 1476719619.997842] Succeeded
cliente hecho

```

Figura 101: Activación del sistema de detección de elementos



Figura 102: Mensaje de alerta 1



Figura 103: Mensaje de alerta 2

Finalmente, al recibir los datos de confirmación de presencia de elementos se inicializa la segunda parte de la secuencia para continuar con el proceso de empernado. La comprobación del funcionamiento de la segunda parte de la secuencia se muestra en la Figura 101. Al finalizar el proceso se informa al usuario que la secuencia ha terminado Figura 105.


```

hay elementos
secuencia
9
[INFO] [WallTime: 1476719619.996943] Moviendo brazos
llego
motor1: 85 motor2: 1025 motor3: 1140 motor4: 700 motor5: 500 motor6: 700 motor7: 822 gripper1: 400
motor11: 715 motor12: 3973 motor13: 2929 motor14: 1330 motor15: 508 motor16: 265 motor17: 514 gripper2: 490
[INFO] [WallTime: 1476719619.997842] Succeeded
cliente hecho
1
[INFO] [WallTime: 1476719629.073866] Moviendo brazos
llego
motor1: 85 motor2: 1025 motor3: 1140 motor4: 700 motor5: 500 motor6: 700 motor7: 822 gripper1: 200
motor11: 715 motor12: 3973 motor13: 2929 motor14: 1330 motor15: 508 motor16: 265 motor17: 514 gripper2: 310
[INFO] [WallTime: 1476719629.075056] Succeeded
cliente hecho
2
[INFO] [WallTime: 1476719638.014853] Moviendo brazos
llego
motor1: 85 motor2: 1025 motor3: 1140 motor4: 600 motor5: 500 motor6: 775 motor7: 822 gripper1: 290
motor11: 680 motor12: 3973 motor13: 2929 motor14: 1433 motor15: 508 motor16: 260 motor17: 514 gripper2: 310
[INFO] [WallTime: 1476719638.015407] Succeeded
cliente hecho
3
[INFO] [WallTime: 1476719647.045238] Moviendo brazos
llego
motor1: 85 motor2: 1025 motor3: 1140 motor4: 600 motor5: 150 motor6: 800 motor7: 822 gripper1: 290
motor11: 690 motor12: 3973 motor13: 2919 motor14: 1460 motor15: 508 motor16: 250 motor17: 514 gripper2: 310
[INFO] [WallTime: 1476719647.046367] Succeeded
cliente hecho
4
[INFO] [WallTime: 1476719656.039107] Moviendo brazos
llego
motor1: 85 motor2: 1025 motor3: 1140 motor4: 600 motor5: 150 motor6: 800 motor7: 822 gripper1: 290
motor11: 680 motor12: 3973 motor13: 2988 motor14: 1080 motor15: 508 motor16: 245 motor17: 514 gripper2: 310
[INFO] [WallTime: 1476719656.040468] Succeeded
cliente hecho

```

Figura 104: Funcionamiento de la secuencia segunda parte



Figura 105: Información, secuencia terminada

El botón parar se programó para detener el sistema en cualquier proceso que se encuentre en ejecución. En la Figura 106 se muestra la cancelación de la secuencia, al presionar el botón parar deja de publicar las siguientes posiciones de la secuencia.

```

secuencia
9
[INFO] [WallTime: 1476661222.806938] Moviendo brazos
llego
motor1: 11 motor2: 21 motor3: 31 motor4: 41 motor5: 51 motor6: 61 motor7: 71 gripper1: 81
motor11: 111 motor12: 211 motor13: 311 motor14: 411 motor15: 511 motor16: 611 motor17: 711 gripper2: 811
[INFO] [WallTime: 1476661222.807832] Succeeded
cliente hecho
[INFO] [WallTime: 1476661223.385972] RobotToolbox state RECALLING when in simple state DONE with SimpleActionClient in /goals
1
[ERROR] [WallTime: 1476661224.186358] SimpleActionClient received DONE twice
2
3
data: True

```

Figura 106: Funcionamiento Parar

4.3. Resumen

En el capítulo desarrollado se describe la implementación del proyecto en base a la estructura modular previamente determinada en función de la estructura, cinemática del manipulador, visión artificial y planificador. Se documentan las hojas de procesos para la manufactura de los elementos mecánicos del torso robótico y se describen los componentes electrónicos implementados contenidos en el mismo. Los planos de construcción de los diseños desarrollados y descritos en el capítulo se encuentran adjuntos como anexos. En el apartado de pruebas y resultados se describe el proceso desarrollado para efectuar las pruebas de funcionamiento asignadas a cada parte del proceso y se exponen los resultados obtenidos y las mejoras obtenidas al final del estudio en cada caso.

La implementación de los componentes mecánicos fue desarrollada a partir del análisis del diseño estático de los mismos, en función de la aplicación y las especificaciones de diseño. La construcción de los elementos cumple con el procedimiento redactado en las hojas de procesos y los planos adjuntos como anexos. Mediante la implementación de la resolución de la cinemática inversa desarrollada en base a algoritmos genéticos se obtuvo un error cuadrático medio menor a 0.1, después de un máximo de 200 iteraciones. En cuanto al desarrollo del tratamiento e implementación de visión artificial, se logró obtener una imagen clara y sin ruido de los elementos dispuestos para el proceso de junta empernada. Se hizo uso de software libre y el cálculo del área y centroide de la imagen de contraste en color verde, de esta forma se identificó la presencia o ausencia de los objetos de la junta como medio de retroalimentación para el desarrollo del proceso automático.

Capítulo 5

Conclusiones, Recomendaciones y Trabajos Futuros

En el capítulo se exponen las conclusiones y recomendaciones obtenidas a partir de del trabajo de titulación realizado. Además, se proponen trabajos futuros que se pueden realizar a partir del presente trabajo.

5.1. Conclusiones

- Se diseñó y construyó un torso colaborativo de dos brazos para la unión de una junta empernada en una celda de trabajo basado en el proyecto AR7D desarrollado por Sánchez y Terán. El manipulador de dos brazos antropomorfos de 7GDL es capaz de desarrollar una junta empernada de forma automática con el uso de software libre y la ayuda de visión artificial.
- Se realizó el diseño mecatrónico de un torso robótico compuesto por dos brazos antropomorfos impresos en 3D y una celda de trabajo para desarrollar un proceso automático de junta empernada. El diseño e implementación del torso se desarrolló en base a las especificaciones geométricas, dimensionales y de resistencia de la aplicación. La estructura del manipulador fue desarrollada en acero ASTM A36, cuyos elementos fueron simulados y analizados para resistir cargas estáticas y volcamiento. Después del estudio realizado, los componentes mecánicos cumplieron con un factor de seguridad superior a 1 en el análisis de cargas y un factor superior a 1.5 en el de volcamiento, cumpliendo con los parámetros de diseño correspondientes en cada caso.
- Se diseñó e implementó una placa PCB que contiene el circuito de potencia alimentado por una fuente de 12 VADC a 6 A y el circuito de control del sistema

determinado por el controlador USB2Dynamixel. La placa está distribuida de forma modular, claramente etiquetadas para ser conectada de forma correcta a cada uno de los componentes y facilitar su mantenimiento. Las conexiones realizadas en la placa corresponden al controlador, la fuente de alimentación y los dos brazos, cada uno con sus respectivos pines. Se realizó también la implementación del panel de control colocado en la parte posterior del manipulador. El panel fue desarrollado en acrílico transparente, el mismo que contiene un interruptor de encendido, un indicador de funcionamiento, un paro de emergencia, un indicador de paro de emergencia y agujeros diseñados a medida para ventilación y disposición de cables USB y de alimentación.

- Se desarrolló e implementó un planificador como unidad de control jerárquica, encargado de administrar y controlar el posicionamiento de cada uno de los brazos. El planificador fue desarrollado como supervisor del proceso en cada una de sus partes, siendo ellas el posicionamiento manual, el desarrollo del proceso automático de junta empernada, “home” y paro de emergencia. Para la resolución de la cinemática inversa del manipulador se hizo uso de algoritmos genéticos y librerías Python. Se implementó una máquina de estados para ejecutar la secuencia asignada al desarrollo del proceso automático de junta empernada. La secuencia parte del primer estado que posiciona ambos brazos en “home”, después procede a situarlos sobre los elementos de la junta para dar paso al proceso de visión artificial, una vez identificados los elementos continúa la secuencia y retornar a la posición inicial al terminar el proceso. Se asignó un botón para llevar los brazos a la posición “home” y adicionalmente se implementó un botón de paro de emergencia para detener cualquier proceso en ejecución.
- Para el desarrollo del proceso automático de junta empernada se realizó la codificación de visión artificial mediante la adquisición y tratamiento de imágenes. Se colocó dos cámaras acopladas a los efectores finales de cada brazo robótico. Los dispositivos son responsables de realizar la realimentación del proceso mediante la detección de los elementos necesarios para realizar la junta empernada.
- Para detectar la presencia de los elementos de la junta empernada se realizaron pruebas de enfoque y selección del rango de colores HSV del perno y la tuerca. Mediante sintonización se logró eliminar el ruido de la imagen y la correcta detección del contraste entre los elementos y la celda de trabajo. Los elementos de la junta fueron impresos en plástico ABS de color verde, de forma que cumple con las especificaciones de contraste, dimensionamiento, geometría y peso necesarios para el desarrollo del proceso automático.

- Se realizaron pruebas de posicionamiento iterativas para generar cada uno de los estados necesarios para el control propuesto. De la misma forma se hicieron pruebas de comunicación entre los nodos del programa para acceder a cada uno de los brazos y posiciones dependiendo del estado en el que se encuentra el proceso. El planificador funciona a manera de supervisor recibiendo y enviando los datos necesarios al manipulador en función del proceso ejecutado desde la interfaz gráfica.

5.2. Recomendaciones y Trabajos Futuros

- Mediante la implementación de un manipulador colaborativo para el desarrollo de aplicaciones industriales, se ve la facilidad de desarrollar procesos complejos simultáneos. Sin embargo, es recomendable hacer un análisis dinámico de los manipuladores y sus actuadores debido a la posición en la que se encuentra colocado el torso y las limitaciones de movilidad debido al volumen de trabajo.
- La estructura del manipulador colaborativo cumple la función de cuerpo, estabilizador y soporte del robot. El volumen de trabajo determinado para la aplicación a desarrollar depende de la posición de los brazos y su disposición. Una de las mejoras que se puede implementar al presente trabajo es realizar un diseño estructural capaz de aumentar el volumen de trabajo del manipulador y reducir el esfuerzo en los actuadores para posiciones críticas.
- Los componentes eléctricos del manipulador fueron dimensionados para trabajar sobre los niveles de consumo máximo de los componentes del sistema. Sin embargo, es recomendable realizar la medición regular de voltaje suministrado por la fuente de alimentación dispuesta en el interior del manipulador, ya que por motivos de manipulación, es posible que el potenciómetro de la misma necesite ser regulado ocasionalmente.
- Para el desarrollo del presente proyecto se utilizó una máquina de estados que setea offline la posición de los actuadores y ejecuta un proceso automático haciendo uso de valores estáticos definidos previamente. Sin embargo, existen varios limitantes para la ejecución del movimiento de la cadena cinemática debido al número de actuadores necesarios para resolver el sistema. Se propone por tanto el desarrollo de un optimizador de trayectoria para generar un desplazamiento más acorde a la complejidad de la aplicación asignada para el manipulador. Además, se propone recurrir a métodos más robustos para la resolución de la ci-

nemática inversa de los manipuladores, como el modelado mediante cinemática diferencial o mapas de Bézier.

- El uso de visión artificial como medio de retroalimentación de un proceso automático como el propuesto en el presente estudio puede ser de gran ayuda en el posicionamiento del manipulador. El filtrado y tratamiento de imágenes puede implementarse en el proceso de detección de objetos para identificar la ubicación y geometría exacta del elemento de forma que optimiza la localización de los mismos. Por esta razón se plantea como trabajo de investigación la generación de un algoritmo capaz de enviar los valores requeridos para la resolución de la cinemática y reposicionamiento del manipulador de manera que el efector final acceda de forma precisa al elemento identificado.
- La detección de los elementos destinados a desarrollar el proceso automático de junta empernada se realizó mediante el cálculo del área de contraste entre la celda de trabajo y el elemento de color. Además, se calculó el centroide del área identificada. Sin embargo, se puede hacer uso de la identificación de bordes y formas de los elementos visualizados por las cámaras para optimizar el posicionamiento de los objetos. Se recomienda también tomar en cuenta los cambios de iluminación en caso de tener problemas con la detección de objetos.
- El programa implementado para resolver la secuencia automática de junta empernada fue desarrollado de forma offline haciendo necesario el posicionamiento iterativo de los manipuladores para generar un movimiento coordinado. El proceso de posicionamiento y orientación de los efectores finales pueden ser realizados y programados haciendo uso de la resolución de la cinemática inversa y optimizadores de trayectoria que permitan definir un proceso online.

Bibliografía

[ABB, 2015] ABB (2015). New abb.

[ABB, 2016] ABB (2016). Abb introduces yumi, world first truly collaborative dual-arm robot.

[Blecha et al., 2008] Blecha, P., Kolíbal, Z., Knoflíček, R., Pochyly, A., Kubela, T., Blecha, R., & Brezina, T. (2008). Módulo 10, robótica.

[Calderón, 2014] Calderón, M. T. (2014). *Control por Visión de un Cuadricóptero Utilizando ROS*.

[Ceccareli, 2013] Ceccareli, M. (2013). Grasping in robotics. *New York*.

[Cepyme, 2016] Cepyme (2016). Guía técnica de seguridad en robótica.

[Chimarro, 2015] Chimarro, J. D. (2015). Diseño, construcción e implementación de un robot esférico de 3 grados de libertad para manipulación de objetos utilizando la plataforma robotic operating system (ros) para la universidad de las fuerzas armadas espe.

[COMERCIO, 2010] COMERCIO, E. (2010). Brazos robóticos a la industria.

[Correa et al., 2011] Correa, J., Vásquez, R., Ramírez, J., & Elkin, T. (2011). Manipuladores robóticos, una mano para la industria.

[Díaz, 2012] Díaz (2012). Despliegue de la función de la calidad.

[EPSON, 2016] EPSON (2016). Autonomous dual-arm robot.

[Gaona & Gallardo, 2014] Gaona, A. & Gallardo, J. (2014). Diseño y construcción de una plataforma robótica móvil para interiores capaz de realizar slam.

[Gayo, 2013] Gayo, J. (2013). Diseño de un brazo industrial neumático. *Universidad Carlos III de Madrid*.

- [Giraldo et al., 2006] Giraldo, L., Delgado, E., & Castellanos, G. (2006). *Cinemática inversa de un brazo robot utilizando algoritmos genéticos*.
- [González, 2004] González, F. A. (2004). *Algoritmos genéticos aplicados al planteamiento de trayectorias de robots móviles, modelamiento y simulación*. Bucaramanga: Universidad Industrial de Santander, Facultad de Ingenierías Fisiomecánicas.
- [Hao et al., 2004] Hao, W., Fuchun, S., Zengqi, S., & Licheng, W. (2004). Optimal trajectory planning of a flexible dual-arm space robot with vibration reduction. *Journal of Intelligent and Robotic Systems*, Kluwer Academic Publishers, 147–163.
- [Herrero et al., 2016] Herrero, H., García, F., Esnaola, U., & Sallé, D. (2016). Aplicaciones de robótica bi-brazo en automatización aeronáutica.
- [Jaramillo, 2005] Jaramillo, A. (2005). *Cinemática de manipuladores robóticos*.
- [Kelly & Santibáñez, 2003] Kelly, R. & Santibáñez, V. (2003). *Control de Movimiento de Robots Manipuladores*. Madrid.
- [Mecalux, 2015] Mecalux (2015). Robot industrial de soldadura.
- [Ministerio de Educación, 2012] Ministerio de Educación, G. d. E. (2012). Visión artificial, aplicación práctica de la visión artificial en el control de procesos industriales.
- [Montes, 2015] Montes, L. (2015). El mundo.
- [MOTOMAN, 2016] MOTOMAN (2016). Yaskawa.
- [OpenCV, 2016] OpenCV (2016).
- [Park et al., 2010] Park, C., Park, K., Il Park, D., & Kyung, J.-H. (2010). Dual arm robot manipulator and its easy. *Department of robotics and intelligent machinery*, (pp. 1–15).
- [Porrás & De la Cruz, 2015] Porrás, J. & De la Cruz, M. (2015). Clasificación system based on computer vision. *Taller de Electrónica IV*.
- [Ramírez, 2012] Ramírez, K. D. (2012). *Cinemática iinversa del robot*.
- [Rivas, 2010] Rivas, H. (2010). *Robótica y automatización industrial*.
- [Robots, 2016] Robots, U. (2016). *La tecnología robótica*.
- [Rojas, 2015] Rojas, M. (2015). *Método en v*.

[Sánchez & Terán, 2015] Sánchez, A. & Terán, A. (2015). Diseño y construcción de un brazo robótico antropomórfico de siete grados de libertad con análisis cinemático y dinámico mediante algoritmos genéricos. *ESPE*.

[TrossenRobotics, 2015] TrossenRobotics (2015). Trossen robótica.

[TrossenRobotics.com, 2016] TrossenRobotics.com (2016). Usb2dynamixel.