

UNIVERSIDAD DE SONORA

DIVISIÓN DE INGENIERÍA

Departamento de Ingeniería Industrial

**REHABILITACIÓN Y ACTUALIZACIÓN DE ROBOT
AMATROL PEGASUS CON SOFTWARE ROS**

TESIS

que para obtener el título de:
INGENIERO EN MECATRÓNICA

PRESENTA:

EDRAÍ VALDIVIEZO COUTIÑO

Hermosillo, Sonora, México

Febrero 2022

Universidad de Sonora

Repositorio Institucional UNISON



**"El saber de mis hijos
hará mi grandeza"**



Excepto si se señala otra cosa, la licencia del ítem se describe como openAccess



UNIVERSIDAD DE SONORA
DIVISION DE INGENIERIA
DEPARTAMENTO DE INGENIERIA
INDUSTRIAL

Hermosillo, Sonora, a 16 de Febrero del 2022

DR. CARLOS FIGUEROA NAVARRO
Coord. del Programa de Ingeniería en Mecatrónica

Por este conducto, hago de su conocimiento que estoy de acuerdo que se realice el examen profesional del alumno EDRAI VALDIVIEZO COUTIÑO con expediente No. 216215675 el cual será el día 4 DE MARZO DE 2022 de manera virtual por la plataforma TEAMS a las 12 p.m.

RELACION DE JURADOS:

NOMBRE	FIRMA
PRESIDENTE: DR. RAFAEL CASTILLO ORTEGA	
SECRETARIO: MC. YNA MARIA DAVILA RASCÓN	
VOCAL: DR. MIGUEL PORCHAS OROZCO	
SUPLENTE: DR. CUITLAHUAC IRIARTE CORNEJO	

ATENTAMENTE

MIEMBROS DEL JURADO

DEDICATORIA

Esta tesis está dedicada a:

A mis padres Emmanuel y Gabriela, quienes con su amor y esfuerzo, pese a cualquier dificultad y circunstancia, me han apoyado en todo momento y por lo cual me han permitido llegar a alcanzar una más de mis metas, les agradezco por alentarme a seguirme esforzando y creer en mí en cada reto que me propongo.

A mi director de tesis y profesor, el Dr. Rafael Castillo Ortega, por creer en mí, por su apoyo aún más allá de su labor como maestro, por su conocimiento e ideas aportadas en la elaboración de esta tesis, siempre le recordaré.

AGRADECIMIENTOS

Agradezco a Dios por permitirme vivir esta experiencia, por darme la sabiduría e inteligencia necesaria para culminar este grado de estudios y por siempre tener cuidado de mí.

Agradezco a mis familiares quienes me apoyaron económicamente para poder sustentar mis estudios en los momentos que lo necesitaba.

Mis más sinceros agradecimientos a mi novia Melissa, quien me ha estado apoyando y motivando a esforzarme a lo largo de estos años.

INDICE DE CONTENIDO

Página

CAPITULO I. ESTADO DEL ARTE	1
1.1 Introducción	2
1.1.1 Los términos Robots Industriales y Robótica	3
1.1.2 Las generaciones de Robots Industriales	4
1.2 Antecedentes históricos del control automático	15
1.2.1 Mecanización y automatización	17
1.3 clasificación de los robots.....	18
1.3.1 Los Robots Móviles	18
1.3.2 Los robots humanoides	19
1.3.3 Los robots industriales.....	21
1.4 Aplicaciones industriales.....	22
1.4.1 Aplicaciones de manipulación.....	23
1.4.2 Aplicaciones por proceso.....	26
1.5 Robótica pedagógica en el aprendizaje de ingeniería	28
1.5.1 ¿Qué es un robot pedagógico?	29
1.5.2 Importancia y beneficios de usar la robótica en la docencia educativa	30
1.5.3 Robots pedagógicos preensamblados y kits.....	31
1.6 Sistema operativo robótico (ROS).....	32
1.6.1 Sistema de Archivos.....	34
1.6.2 Computación a nivel gráfico.....	35
1.6.3 Distribuciones de ROS	36
1.6.4 ROS y la docencia.....	38
CAPITULO II. GENERALIDADES.....	40
2.1 Introducción	40
2.2 Morfología de los robots.....	40

2.2.1 Grado de libertad	40
2.2.2 Tipos de articulaciones	41
2.2.3 Estructuras de robots manipuladores.....	43
2.2.4 Estructuras básicas.....	44
2.2.5 Configuración Cartesiana.....	44
2.2.6 Configuración Cilíndrica	46
2.2.7 Configuración polar o esférica.....	48
2.2.8 Configuración SCARA.....	50
2.2.9 Configuración angular	51
2.2.10 Configuración paralela	53
2.3 Sensores	55
2.3.1 Sensor óptico reflectivo.....	55
2.4 Servomotores	56
2.4.1 Constitución del servomotor	56
2.4.2 Tipos de controladores.....	57
2.4.3 Funcionamiento del controlador del servomotor	58
2.5 Sistema de Control.....	62
2.5.1 Control de posición por PWM.....	62
2.6 Consideraciones en el diseño mecánico.....	63
2.6.1 Consideraciones técnicas de diseño	63
2.6.2 Materiales para engranes	64
2.6.3 Grafica de esfuerzo-deformación del acero y aluminio.....	65
2.6.4 Tipos de actuadores	67
2.6.5 Transmisión de potencia	69
2.6.6 Ejes.....	69
2.6.7 Engranes	70
2.6.8 Acoplamientos	72

2.7 Subsistemas	74
2.7.1 Subsistema de la base.....	74
2.7.2 Subsistema del hombro	76
2.7.3 Subsistema del codo.....	77
2.7.4 Subsistema de la muñeca.....	78
2.7.5 Subsistema del gripper	78
2.7.6 Subsistema del controlador.....	79
CAPITULO III. DISEÑO	80
3.1 Introducción	80
3.2 Interfaz ROS-Arduino	80
3.2.1 Instalación y configuración de interfaz ROS.....	80
3.2.2 Configuración del Arduino serial	83
3.4 Tarjeta madre.....	89
3.5 Etapa de potencia.....	90
CAPITULO IV. EVALUACIÓN ECONÓMICA.....	92
4.1 Introducción	92
4.2 Costos indirectos	92
4.3 Costos directos o variables.....	92
4.4 Costos y presupuestos.....	93
4.4.1 Componentes electrónicos.....	93
4.4.2 Componentes mecánicos.....	94
4.4.3 Costos mano de obra e ingeniería	95
4.4.4 Conjunto de costos	97
4.4.5 Comparativa con otras opciones del mercado	98
CAPITULO V. RESULTADOS.....	99
Introducción	99
5.1 Sistema electrónico.....	99

5.2 Sistema eléctrico	101
5.3 Sistema mecánico	103
5.4 Sistema de control	104
CAPITULO VI. CONCLUSIONES.....	106
Referencias	108
ANEXOS	113

ÍNDICE DE FIGURAS

Página

FIGURA 1. 1 JOSEPH ENGELBERGER Y GEORGE DEVOL. [14] ROBOT UNIMATE. [14].....	FIGURA 1. 2 EL 6
FIGURA 1. 3 EL ROBOT VERSATRAN. [16]	7
FIGURA 1. 4 BRAZO ROBÓTICO DE STANFORD. [16].....	9
FIGURA 1. 5 ROBOT UNIMATE PUMA. [16].....	10
FIGURA 1. 6 EL "LEGENDARY" ABB IRB ROBOT. [16]	10
FIGURA 1. 7 UNO DE LOS PRIMEROS PROTOTIPOS DE ROBOT SCARA, DISEÑADO POR HIROSHI MAKINO. [16].....	12
FIGURA 1. 8 EJEMPLOS DE ADEPTONE DE SCARA ROBOTS. [16].....	13
FIGURA 1. 9 ROBOTS DELTA EN LAS CELDAS DE TRABAJO DE DEMAUREX PARA EL EMPAQUETAMIENTO DE PRETZELES. [16].....	14
FIGURA 1. 10 EL ROBOT FLEXPICKER DE ABB. [16].....	14
FIGURA 1. 11 EJEMPLO DE DIAGRAMA DE CONTROL. AUTORÍA PROPIA.	16
FIGURA 1. 12 ROBOTS MASCOTAS EN LA HISTORIA. [25].....	19
FIGURA 1. 13 EJEMPLOS DE ROBOTS HUMANOIDES EN EL SIGLO XXI. [26].....	20
FIGURA 1. 14 ROBOTS EN LÍNEA DE ENSAMBLAJE DE FÁBRICA DE AUTOS POR ALEJANDRO SERRANO.....	21
FIGURA 1. 15 ROBOT MANIPULADOR PEGASUS DE AMATROL.	25
FIGURA 1. 16 ROBOT IRB 910SC DE ABB TIPO SCARA.....	25
FIGURA 1. 17 ROBOT KR QUANTEC PARA SOLDADURA POR PUNTOS DE KUKA.	26
FIGURA 1. 18 ROBOT ECO RP E043I PARA PINTURA AUTOMOTRIZ DE DÜRR.	27
FIGURA 1. 19 ROBOT ARTICULADO MC2000 II DE YASKAWA.....	27
FIGURA 1. 20 ROBOT EDUCACIONAL THYMIO DE MOBSYA. FIGURA 1. 21 ROBOT EDUCACIONAL DASH DE WONDER WORKSHOP.....	31
FIGURA 1. 22 KIT EDUCACIONAL LEGO MINDSTORMS FIGURA 1. 23 KIT DE BRAZO ROBOTICO POPPY ERGO JR	32
FIGURA 1. 24 MAPA DE USUARIOS DE ROS EN EL MUNDO [29].....	39
FIGURA 2. 1 ILUSTRACIÓN DE GRADOS DE LIBERTAD DE UN NODO. [35]	41
FIGURA 2. 2 A) ARTICULACIÓN ROTACIONAL (1 GDL) . B) ARTICULACIÓN CILÍNDRICA (2 GDL) C) ARTICULACIÓN PRISMÁTICA (1 GDL)	42

FIGURA 2. 3 A) ARTICULACIÓN ESFÉRICA (3 GDL)	B) ARTICULACIÓN PLANAR (2GDL) C) TORNILLO (1GDL)	43
FIGURA 2. 4 EJEMPLO DE ESTRUCTURA DE ROBOT MANIPULADOR. [36].....		43
FIGURA 2. 5 VOLUMEN DE TRABAJO DE ROBOTS MANIPULADORES. [22].....		44
FIGURA 2. 6 ESTRUCTURA DE ROBOT TIPO CARTESIANO. [35]		45
FIGURA 2. 7 ROBOT CARTESIANO DE 3 EJES EPSON'S RP-HMSz		45
FIGURA 2. 8 VOLUMEN DE TRABAJO DE ROBOT TIPO CARTESIANO. [35].....		46
FIGURA 2. 9 ROBOT DE CONFIGURACIÓN TIPO CILÍNDRICO. [35]		46
FIGURA 2. 10 VOLUMEN DE TRABAJO DE ROBOT TIPO CILÍNDRICO. [35].....		47
FIGURA 2. 11 ROBOT DE CONFIGURACIÓN CILÍNDRICA MODELO RT3200 POR SEIKO.....		47
FIGURA 2. 12 ESTRUCTURA DE ROBOT DE CONFIGURACIÓN POLAR. [35]		48
FIGURA 2. 13 VOLUMEN DE TRABAJO DE ROBOT TIPO POLAR.		49
FIGURA 2. 14 ROBOT DE CONFIGURACIÓN POLAR DESARROLLADO POR LA COMPAÑÍA UNIMATION. [14].....		49
FIGURA 2. 15 ESTRUCTURA DE ROBOT TIPO SCARA. [35].....		50
FIGURA 2. 16 VOLUMEN DE TRABAJO DE ROBOT TIPO SCARA. [35]		50
FIGURA 2. 17 ROBOT TIPO SCARA DESARROLLADO POR LA COMPAÑÍA IXA.		51
FIGURA 2. 18 ESTRUCTURA DE ROBOT CON CONFIGURACIÓN ANGULAR. [35].....		52
FIGURA 2. 19 VOLUMEN DE TRABAJO DE ROBOT CON CONFIGURACIÓN ANGULAR. [35]		52
FIGURA 2. 20 ROBOT PEGASUS DESARROLLADO POR AMATROL.		53
FIGURA 2. 21 ESTRUCTURA DE ROBOT CON CONFIGURACIÓN PARALELA. [37].....		53
FIGURA 2. 22 VOLUMEN DE TRABAJO DE ROBOT CON CONFIGURACIÓN PARALELA. [37]		54
FIGURA 2. 23 ROBOT DELTA OMRON PARALELO POR LA COMPAÑÍA ADEPT.....		54
FIGURA 2. 24 DIAGRAMA DE UN SENSOR OPTOREFLECTIVO DE HERRADURA.		55
FIGURA 2. 25 DIAGRAMA DE BLOQUE DE SERVOMOTOR DIGITAL. [39]		59
FIGURA 2. 26 DIAGRAMA DE DIMENSIONAMIENTO MECÁNICO DEL SERVOMOTOR 5160SSG MARCA DS.		61
FIGURA 2. 27 DIAGRAMA DE DIMENSIONAMIENTO MECÁNICO DEL SERVOMOTOR 8120MG MARCA READYTOSKY		61
FIGURA 2. 28 CONTROL DE UN SERVOMOTOR MEDIANTE PWM. AUTORÍA PROPIA.		62
FIGURA 2. 29 GRAFICA DE ESFUERZO-DEFORMACIÓN DEL ALUMINIO 6020-T6 [42]		66
FIGURA 2. 30 GRAFICA DE ESFUERZO-DEFORMACIÓN DEL ACERO AISI1020 [43].....		66
FIGURA 2. 31 REPRESENTACIÓN NORMALIZADA DE ENGRANAJES Y PARÁMETROS CARACTERÍSTICOS. AUTOR: MANUEL TORRES BÚA.....		70

FIGURA 2. 32 ENGRANAJE RECTO [45].	71
FIGURA 2. 33 ENGRANAJES HELICOIDALES [45].	71
FIGURA 2. 34 ENGRANAJES CÓNICOS [45].	72
FIGURA 2. 35 ACOPLAMIENTOS RÍGIDOS DE BRIDA 8MM. AUTORÍA PROPIA.	73
FIGURA 2. 36 FIJADOR DE TORNILLOS Y TUERCAS UTILIZADO EN EL PROYECTO. AUTORÍA PROPIA.	74
FIGURA 2. 37 SUBSISTEMA DE LA BASE. AUTORÍA PROPIA.	75
FIGURA 2. 38 MECANISMO DEL SUBSISTEMA DE LA BASE. AUTORÍA PROPIA.	75
FIGURA 2. 39 PLACA INSTALADA EN LA BASE PARA EL CABLEADO. AUTORÍA PROPIA.	76
FIGURA 2. 40 SUBSISTEMA DEL HOMBRO. AUTORÍA PROPIA.	76
FIGURA 2. 41 ILUSTRACIÓN DE ACOPLAMIENTO RÍGIDO INSTALADO EN EL ROBOT. AUTORÍA PROPIA	77
FIGURA 2. 42 SUBSISTEMA DEL CODO. AUTORÍA PROPIA.	77
FIGURA 2. 43 SUBSISTEMA DE LA MUÑECA. AUTORÍA PROPIA.	78
FIGURA 2. 44 SUBSISTEMA DEL GRIPPER. AUTORÍA PROPIA.	79
FIGURA 3. 1 ESQUEMA DE TARJETA MADRE. AUTORÍA PROPIA.	90
FIGURA 3. 2 TARJETA MADRE DEL SUBSISTEMA DE CONTROL. AUTORÍA PROPIA.	90
FIGURA 5. 1 GABINETE DE CONTROL ORIGINAL DEL ROBOT PEGASUS DE AMATROL.	100
FIGURA 5. 2 GABINETE HABILITADO PARA EL CONTROL DEL ROBOT PEGASUS DE AMATROL.	100
FIGURA 5. 3 FUENTE DE POTENCIA ORIGINAL DEL ROBOT PEGASUS DE AMATROL.	101
FIGURA 5. 4 FUENTE POTENCIA UTILIZADA EN LA RESTAURACIÓN DEL ROBOT PEGASUS.	102
FIGURA 5. 5 ADAPTADOR DE CORRIENTE MARCA SONY UTILIZADO EN LA RESTAURACIÓN DEL ROBOT PEGASUS.	102
FIGURA 5. 6 REGULADOR DE VOLTAJE BUCK INSTALADO EN PLACA MADRE.	103
FIGURA 5. 7 PLANOS DE DISEÑO DE SOPORTE PARA SERVOMOTOR. AUTORÍA PROPIA.	104
FIGURA 5. 8 A) VISTA LATERAL B) VISTA FRONTAL DEL ROBOT PEGASUS.	104
FIGURA A 1 MODELO EN 3D DEL ROBOT PEGASUS REALIZADO EN SOLIDWORKS. AUTORÍA PROPIA.	113
FIGURA A 2 UBICACIÓN DEL ESPACIO DE TRABAJO DE CATKIN. AUTORÍA PROPIA.	113
FIGURA A 3 CONSOLA CON EL ESPACIO DE TRABAJO CATKIN. AUTORÍA PROPIA.	114

FIGURA A 4 EL ASISTENTE DE MOVEIT!. AUTORÍA PROPIA.	114
FIGURA A 5 DIVISIÓN DEL MODELO DE ROBOT EN GRUPOS. AUTORÍA PROPIA.....	115
FIGURA A 6 DEFINIENDO POSES PREESTABLECIDAS EN EL ROBOT. AUTORÍA PROPIA.....	115
FIGURA A 7 CONFIGURANDO LOS CONTROLES DE ROS PARA EL ROBOT. AUTORÍA PROPIA...	116
FIGURA A 8 MENÚ DE PLANEACIÓN DE MOVIMIENTOS MOVEIT!. AUTORÍA PROPIA.....	117
FIGURA A 9 MENÚ DE CONTROL INDIVIDUAL DE LOS EJES DEL ROBOT. AUTORÍA PROPIA.	117
FIGURA A 10 COORDENADAS DE POSICIÓN DEL ROBOT.	118

ÍNDICE DE TABLAS

TABLA 1.1 CLASIFICACIÓN DE ROBOTS.....	18
TABLA 1.2 APLICACIONES DE ROBOTS INDUSTRIALES.....	23
TABLA 2.1 ESPECIFICACIONES TÉCNICAS DE SERVOMOTOR 60KG 5160SSG MARCA DS.....	60
TABLA 2.2 ESPECIFICACIONES DE OPERACIÓN DE SERVOMOTOR 60KG 5160SSG MARCA DS. 60	
TABLA 2.3 ESPECIFICACIONES DE OPERACIÓN DE SERVOMOTOR 35KG 8120MG MARCA READYTOSKY	61
TABLA 2.4 COMPARATIVA DE ACTUADORES [44].	69
TABLA 4.1 COSTOS DEL MATERIAL PARA ELABORACIÓN DE TARJETA MADRE. PRECIOS (2021).94	
TABLA 4.2 COSTOS DE COMPONENTES MECÁNICOS PARA EL ROBOT. PRECIOS (2021).....	95
TABLA 4.3 COSTOS DE MANO DE OBRA POR FABRICACIÓN DE TARJETA MADRE.	96
TABLA 4.4 COSTOS DE MANO DE OBRA POR REEMPLAZO DE MOTORES.....	96
TABLA 4.5 COSTOS DE MANO DE OBRA POR RESTAURACIÓN DEL ROBOT PEGASUS.	97
TABLA 4.6 COSTO TOTAL DE LA MANO DE OBRA DEL PROYECTO.....	97
TABLA 4.7 COSTO TOTAL DE LA REALIZACIÓN DEL PROYECTO DE RESTAURACIÓN.	97
TABLA 4.8 COMPARACIÓN ENTRE OPCIONES DE RESTAURACIÓN. PRECIOS (2021).....	98

RESUMEN

El siguiente proyecto de tesis consiste en la rehabilitación y modernización del robot "Pegasus" desarrollado por la compañía AMATROL, este es un brazo robot de uso didáctico comúnmente usado para introducir a la robótica a estudiantes de ingeniería.

Los principales objetivos son rehabilitar el robot en su total funcionamiento, integrar componentes de hardware y software para así tener una herramienta de aprendizaje acorde a las tecnologías de hoy en día, tener conocimiento de las tendencias actuales en la industria manufacturera de robots manipuladores y una introducción al control y programación de robots por medio del software Robot Operating System (ROS).

En este proyecto se presentan aspectos como la modelación en 3D del robot, diseño cinemático para delimitar uniones y grados de libertad, morfología de los brazos manipuladores y uso de software para el control y automatización de tareas. Por otra parte, se adaptaron las partes mecánicas nuevas en el chasis del brazo manipulador, estas vendrían siendo los actuadores los cuales fueron servomotores que remplazaron a motores de corriente directa (DC) que trabajaban con base en enconder.

Al trabajar en la rehabilitación de una máquina al igual que en el desarrollo de un proyecto es necesario analizar tres puntos muy importantes los cuales incluyen ingeniería, economía y ergonomía. La factibilidad del proyecto depende de la confluencia de estos; se llevó un análisis de cada uno los cuales se desarrollan en los capítulos de Diseño y Evaluación económica. Finalmente se expone el resultado de la rehabilitación del brazo manipulador, haciendo una comparativa del estado en que se acogió y el estado actual, así como sus posibles aplicaciones y futuras actualizaciones que se le pudiesen dar.

JUSTIFICACIÓN

En el documento Educación, Metas Educativas 2021 [1, pp. 102-141], se abordan las nuevas tecnologías en educación, y el impacto de dichas tecnologías en la vida cotidiana.

Desde 1976, varios investigadores, principalmente de Canadá y Francia, han estado trabajando en lo que denominaron robótica pedagógica, la cual hoy en día contribuye a la educación en México con elementos multidisciplinares de la robótica, con todo tipo de herramientas didácticas, contribuyen al desarrollo de habilidades como el aprendizaje de las matemáticas, la informática, las ciencias y la introducción a la tecnología, entre otros temas.

La robótica en la educación ha tenido un impacto positivo, porque desarrolla competencias deseables, como el trabajo en equipo, un nivel de comprensión competitivo de la programación y electrónica. Por ejemplo, se requiere procesar alguna información, ya sea usando una computadora o un dispositivo móvil, y esto implica el uso de protocolos de comunicación. Mientras tanto, para el control de los mecanismos, también es importante el conocimiento de la informática, las matemáticas, mecánica y electrónica, entre otros.

En la Universidad de Sonora en el laboratorio de Robótica se cuenta con tres Robots marca Fanuc que fueron donados por Ford Motor Company en el año 2016 y estos fueron puestos en operación en el 2017 por estudiantes de la carrera de ingeniería en Mecatrónica, han sido de gran beneficio para la comunidad estudiantil de ingeniería ayudando a actualizar sus conocimientos en manipulación de robots industriales y aportando gran valor al perfil del ingeniero egresado ya que haber realizado practicas físicas con estos robots es una experiencia y conocimiento irremplazable.

Dado las premisas anteriores, es necesario destacar que no se cuenta con un robot manipulador en la Universidad de Sonora que su sistema sea seguro, sencillo de configurar y modificar, con este robot manipulador PEGASUS se permite al alumno ver directamente el hardware y cada componente que este lo conforma, sin olvidar que el control de este robot introduce a los estudiantes al

Sistema Operativo de Robots (ROS) el cual es robusto y de código abierto. Con esto el alumnado de la Universidad de Sonora se ve beneficiado de tener acceso a esta herramienta de aprendizaje y practica para futuros proyectos de robótica que puedan elaborar en la industria.

OBJETIVO GENERAL

La rehabilitación, restauración, mantenimiento y puesta en operación del robot manipulador Amatrol Pegasus por medio del software ROS. El robot deberá ser capaz de mover de manera satisfactoria objetos de un peso no mayor a 250 gramos, simulando estar en una planta de producción. La programación de coordenadas y rutinas para que sean ejecutadas por el robot deben ser programadas en el software ROS. Con la rehabilitación de dicho robot se busca utilizarlo como herramienta didáctica en el laboratorio de Robótica.

OBJETIVOS ESPECIFICOS

- Rehabilitación con mantenimiento correctivo de componentes dañados.
- Mantenimiento preventivo en los mecanismos del robot.
- Habilidad física del robot manipulador por medio de nuevos motores y componentes electrónicos para que pueda acatar los comandos dados en una computadora.
- Instalación del espacio de trabajo de ROS en una computadora para programar el robot.
- Programar una rutina donde demuestre el robot que puede manipular y transportar objetos.
- Legar el robot, herramientas, investigación y equipo a estudiantes para ser utilizado con fines didácticos en el laboratorio de Robótica.

CAPITULO I. ESTADO DEL ARTE

El diseño de diseñar y construir algún tipo de ser, dispositivo o mecanismo que pueda realizar tareas repetitivas o pesadas data desde la antigüedad. Desde la época greco-helenística, algunos de estos dispositivos, que recibieron el nombre de autómatas, han sido diseñados y creados por ingeniosos inventores, pertenecientes a diferentes civilizaciones a lo largo de los siglos. El origen del término “robot” se sitúa en tiempos más recientes: es decir, proviene de la palabra checa “robota”, que significa “trabajo pesado” o “trabajo forzoso” [2]. La introducción de este término se debe al escritor checo Karel Čapek (1890-1938), quien lo utilizó por primera vez en 1920 en su novela “R.U.R. : Rossum’s Universal Robots” [3].

Las aplicaciones industriales de la robótica adquirieron una importancia primordial en el último siglo. El inicio de la “Robótica Industrial”, como la definimos actualmente, se remonta a la década de 1950, aunque algunos tipos de automatización en el entorno industrial comenzaron a aparecer desde los tiempos de la Revolución Industrial.

Los principales objetivos de la robótica industrial son: aumentar la productividad, la cual fundamentalmente se logra optimizando la velocidad de desplazamiento del robot manipulador, con lo cual aumenta el rendimiento total de la línea de producción, mejorar la calidad de los productos fabricados por medio de la instrumentación de un control de calidad, optimizar el rendimiento de otras máquinas, herramientas relacionadas a la producción y el realizar trabajos en condiciones hostiles y peligrosas.

1.1 Introducción

El objetivo de este capítulo es sentar las bases sobre el desarrollo del robot industrial e introducir a los robots pedagógicos. Se discuten las características de los robots industriales y la robótica, abarcando lo que es la automatización, las diferentes disciplinas que dan paso a la robótica y la mecánica histórica de los robots.

Antes de hablar de los antecedentes de la robótica es necesario dar una introducción a este concepto, iniciando por las leyes fundamentales de la robótica. El conocido escritor ruso de ciencia ficción Isaac Asimov formuló las tres leyes fundamentales para los robots [4]. La perspectiva aquí es, en contraste con el robot descrito por Čapek, un robot benevolente y bueno que sirve el ser humano. Asimov visualizó al robot como una criatura mecánica (autómata) de apariencia humana sin sentimientos. El comportamiento es dictado por un "cerebro" programado por los seres humanos, de tal forma que se cumplieran determinadas reglas éticas.

El término robótica fue introducido posteriormente por Asimov, como la ciencia dedicada al estudio de los robots la cual se basó en las leyes de la robótica. Estas tres reglas fueron complementadas por la ley cero de Asimov en 1985. Desde el establecimiento de las leyes de los robots, la palabra robot ha alcanzado el significado alternativo como un producto industrial diseñado por ingenieros o técnicos especializados [5, 6].

Las leyes fundamentales de la robótica:

0. Un robot no puede hacer daño a la humanidad o, por inacción, permitir que la humanidad sufra daño [6].
1. Un robot no dañará a un humano o por no realizar una acción permitirá que el humano sufra daño alguno.
2. Cumplir las órdenes impartidas por un humano es el deber principal de un robot, menos las que entren en conflicto con la primera ley.

3. Es necesario que el robot proteja su existencia, sin que entre en conflicto con la primera o la segunda ley dicha protección [4].

1.1.1 Los términos Robots Industriales y Robótica

La distinción de los robots radica de cierta forma en la sofisticación de la programabilidad del dispositivo: una fresadora de control numérico (NC) no es un robot industrial. Como dice John J. Craig [7, p. 3], “si un dispositivo mecánico puede programarse para realizar una amplia variedad de aplicaciones, probablemente sea un robot industrial”. La diferencia esencial entre un robot industrial y una máquina NC es la versatilidad del robot, que es provisto con diferentes tipos de herramientas y tiene un gran espacio de trabajo en comparación con el volumen del propio robot. La máquina NC está dedicada a una tarea especial, aunque de una forma bastante flexible, de manera que, da un sistema construido según especificaciones fijas y limitadas [5].

El estudio y control de robots industriales no es una ciencia nueva, sino una mezcla de los “campos clásicos”. Desde la ingeniería mecánica que estudia la estática de la maquina y situaciones dinámicas hasta las matemáticas que permiten describir los movimientos espaciales. Por medio de la teoría del control se proporcionan herramientas para diseñar y evaluar algoritmos para lograr el movimiento deseado. La ingeniería electrónica es útil al diseñar sensores e interfaces para robots industriales. Por último, pero no menos importante, la informática la cual proporciona la programación del dispositivo para realizar una tarea deseada o dotarlo de ciertas habilidades como la inteligencia artificial (IA) [7].

El término robótica se ha definido recientemente como la ciencia que estudia “la conexión inteligente de la percepción y la acción” [5]. La robótica industrial es una disciplina concerniente al diseño, control y aplicaciones de los robots en la industria. El estado de la tecnología robótica puede reflejarse en la definición de robot procedente del Robot Institute of America, el instituto maneja la definición de que "un robot es un manipulador multifuncional reprogramable diseñado para

mover materiales, piezas, herramientas o dispositivos especializados a través de movimientos programados variables para la realización de una variedad de tareas” [5, p. 4]. El elemento clave en la definición es la palabra reprogramable, que le da al robot características como utilidad y adaptabilidad. A veces se menciona la palabra revolución robótica, pero de hecho es parte de la revolución informática más grande [8].

La mayoría de las organizaciones hoy en día asienten con la definición de robot industrial, formulado por la Organización Internacional de Normalización, ISO:

- Robot Manipulador industrial es un autómata controlado, reprogramable, multipropósito, máquina manipuladora con varios grados de libertad, los cuales pueden ser fijos o móviles para su uso en aplicaciones de automatización industrial [9, p. 25].
- El manipulador es una máquina, cuyo mecanismo suele constar de una serie de segmentos articulados o deslizantes entre sí, con el fin de agarrar y/o mover objetos (piezas o herramientas) generalmente en varios grados de libertad [9, p. 25].

Se puede observar a partir de las definiciones anteriores que la palabra manipulador se utiliza para nombrar el brazo del robot. La definición de robot industrial se puede interpretar de la siguiente manera: un robot debe ser fácilmente reprogramable sin reconstruir físicamente la máquina. También tendrá memoria y lógica para poder trabajar de forma autónoma. Esta estructura mecánica debe poder utilizarse en varias tareas de trabajo sin necesidad de una modificación mecánica grande en su estructura.

1.1.2 Las generaciones de Robots Industriales

La primera generación de robots industriales se extiende desde 1950 hasta 1967. Los robots de esta generación eran básicamente máquinas programables que no tenían la capacidad de controlar realmente la modalidad de ejecución de la tarea; además, no tenían comunicación con el entorno externo. Con respecto al

hardware, los robots de primera generación contaron con equipos de baja tecnología y los servocontroladores no estaban presentes [10]. Una característica peculiar de estos robots es el fuerte ruido que producían cuando sus brazos chocaban con los topes mecánicos contruidos para limitar el movimiento de los ejes. Con respecto a los actuadores, casi todos los robots de primera generación empleaban actuadores neumáticos y estaban controlados por una especie de "puertas lógicas" que actuaban como reguladores automáticos. Tales "puertas lógicas" eran básicamente levas que activaban válvulas neumáticas o relés que controlaban válvulas solenoides. Finalmente, las tareas que los robots de primera generación eran capaces de realizar eran bastante triviales, como la carga-descarga o las simples operaciones de manipulación de materiales.

La historia de la robótica industrial está ambientada convencionalmente en la década de 1950, aunque antes se habían producido algunos avances en la automatización como: el nombrado, dispositivo de pulverización de pintura "programable" inventado por Pollard y Roselund en 1938 [11], y un dispositivo teleoperado "Manipulador" inventado por Goertz en 1949, por dar ejemplo. Sin embargo, el punto de inflexión para la robótica industrial se debió al genio de George Devol, quien diseñó en 1954 una "Transferencia de artículo programable" (este fue el nombre que se le dio cuando se presentó la solicitud de patente [12]. Tal dispositivo fue la base para el desarrollo de Unimate, que se considera el primer robot industrial "verdadero" de la historia.

Devol [13], que era básicamente científico, necesitaba un compañero emprendedor con quien plasmar de manera más concreta su idea de un manipulador robótico que pudiera utilizarse para automatizar procesos industriales. En 1954 conoció a Joseph Engelberger (Figura 1.1), un ingeniero empleado en la industria espacial, y acordaron crear una empresa que pudiera fabricar robots para aplicaciones industriales. Esto llevó a la fundación de una empresa llamada Unimatron, que produjo en 1961 el primer robot Unimate (Figura 1.2). Unimate, fue un robot que se accionaba hidráulicamente, se instaló

inmediatamente en una empresa automotriz, concretamente en la fábrica de General Motors ubicada en Trenton (Nueva Jersey, EE. UU.), donde se empleó para extraer piezas de una máquina de fundición a presión (como se mencionó anteriormente, se utilizó para una sola tarea, porque era muy complicado reprogramarlo). En los años siguientes, se instalaron varios otros Unimates en fábricas de automóviles, donde se emplearon principalmente para la soldadura por puntos de automóviles y para la manipulación de piezas de trabajo [10].



Figura 1. 1 Joseph Engelberger y George Devol. [14]

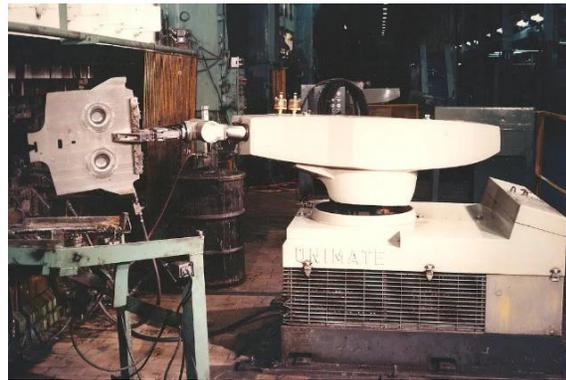


Figura 1. 2 El robot UNIMATE. [14]

Otra de las empresas pioneras en la robótica fue, AMF Corporation: en 1962 fabricaron un nuevo robot que se llamó Versatran (es decir, "transferencia versátil"). Se trataba de un robot cilíndrico (Figura 1.3) que fue encargado por Ford para sus plantas de producción en Canton (Ohio, EE. UU.), gozando así de una buena popularidad [14]. El Versatran también fue el primer robot que se instaló en un sitio productivo en Japón (1967). En 1969, la presencia de robots en empresas japonesas tuvo un aumento significativo, también debido a que Unimation otorgó a Kawasaki Heavy Industries Ltd. la licencia para producir robots para el mercado japonés y asiático. En particular, esto llevó al desarrollo del Kawasaki-Unimate 2000, el primer robot industrial jamás construido en Japón.

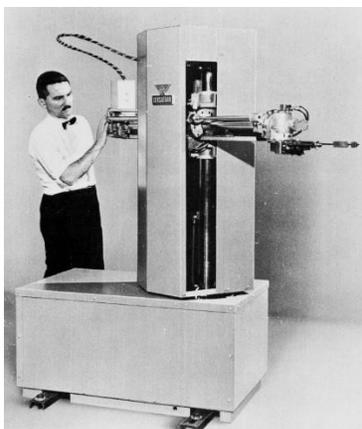


Figura 1. 3 El robot Versatran. [16]

La difusión de los robots en Europa tuvo lugar más o menos al mismo tiempo que en Japón, es decir, en la segunda mitad de la década de 1960. La primera instalación robótica en Europa apareció en 1967 en la empresa Svenska Metallverken ubicada en Upplands Väsby (Suecia): consistía en robots que realizaban tareas sencillas de pick-and-place [10]. Dos años más tarde, en Noruega, el primer robot de pintura fue fabricado por la empresa Tralffa y se instaló en una planta de pintura para carretillas.

La aparición de robots permitió automatizar importantes tareas de producción como la soldadura. Unimation fue la primera empresa en fabricar robots de soldadura, mientras que General Motors fue la primera empresa en instalar este tipo de dispositivos robóticos en sus plantas automotrices de Lordstown (Ohio, EE. UU.) En 1969. Tres años después, FIAT instaló los primeros robots de soldadura en Europa. en sus plantas de Turín (Italia).

Los robots industriales de la segunda generación (que van convencionalmente desde 1968 a 1977) eran máquinas programables básicas con posibilidades limitadas de comportamiento auto adaptativo y capacidades elementales para reconocer el entorno externo [15]. Estos robots usaban servocontroladores, lo que les permitía realizar movimientos de punto a punto y también rutas continuas. Su sistema de control constaba de microprocesadores o de Controladores Lógicos Programables (PLC), y también podían ser programados por un operador

mediante una caja de aprendizaje. Con respecto a los pertenecientes a la primera generación, estos robots podrían realizar tareas más complejas (por ejemplo, control de centros de trabajo). Sin embargo, su nivel de versatilidad no era muy alto, porque cada robot tenía su propio software, que estaba dedicado a una tarea específica. Por lo tanto, estos robots resultaron ser dispositivos de aplicación específica, lo que significa que era muy difícil emplear el mismo robot para diferentes tareas, ya que esto requeriría una modificación sustancial del controlador y una reprogramación completa del software operativo. En cuanto a los diagnósticos que podía llevar a cabo el robot en el sistema, los robots de la segunda generación fueron poco eficientes, ya que los únicos informes de diagnóstico que pudieron producir fueron los relacionados con fallas, los cuales eran reportados por medio de luces indicadoras, sin dar ningún indicio relacionado con la causa de la falla al operador para rastrear.

Al comienzo de la historia de la robótica industrial, los robots tenían actuadores hidráulicos. El cambio de actuadores hidráulicos a eléctricos tuvo lugar en la década de 1970, cuando los componentes electrónicos necesarios para gobernar un robot alcanzaron la madurez técnica completa. De hecho, los microprocesadores y otros componentes comenzaron a ser ampliamente utilizados en ese momento: esto permitió a los fabricantes de robots disponer de dispositivos potentes y rentables que podrían emplearse para implementar sistemas de control capaces de hacer frente a una tarea compleja y computacionalmente costosa, como el control de un robot.

Desde el punto de vista científico, una base significativa para el desarrollo de robots eléctricos fue el mérito de Víctor Scheinman [16]. Scheinman era un estudiante de ingeniería mecánica en la Universidad de Stanford que en 1969 diseñó y construyó el famoso Stanford Arm (Figura 1.4). Este robot fue el primer prototipo de un robot accionado por motores eléctricos (6 motores de CC) y controlado por un microprocesador PDP-6. El brazo de Stanford tenía 5 juntas revolucionarias y una prismática, para un total de 6 DOF, y su cadena cinemática

estaba hecha de transmisiones armónicas y reductores de engranajes rectos. Su cinemática inversa se pudo resolver analíticamente en forma cerrada, lo que permitió una rápida ejecución de la trayectoria, en otras palabras, se pudo sintetizar las juntas activas y pasivas en modo de variables que estaban en función de las coordenadas del efector final, para ello se montaron en el robot algunos sensores (tacómetros y potenciómetros) para medir la posición y la velocidad de las articulaciones del robot.



Figura 1. 4 Brazo robótico de Stanford. [16]

Cuatro años después (1973), Scheinman diseñó otro robot eléctrico, llamado Vicarm, que era más pequeño y liviano que los robots industriales de esa época. Esto hizo que Vicarm fuera especialmente adecuado para su uso en tareas, como el montaje de piezas, en las que no se requería que el robot levantara y transportara cargas pesadas. Estas características de Vicarm fueron tan apreciadas que Unimation compró la empresa que produjo Vicarm y aprovechó su know-how para diseñar y fabricar (en 1978) el famoso robot PUMA (Figura 1.5).



Figura 1. 5 Robot UNIMATE PUMA. [16]

El PUMA (un acrónimo de Programmable Universal Machine for Assembly) fue considerado durante muchas décadas el arquetipo de los robots antropomórficos, y su cinemática todavía se toma como ejemplo en varios libros de robótica a nivel de pregrado y posgrado.

En 1974, la empresa sueca ASEA (ahora ABB) inició la producción de los robots de la famosa y exitosa serie IRB, muy conocida en todo el mundo también por su típico color naranja. El primer robot de esta serie, que se fabricó durante más de 20 años, fue el IRB-6, que se empleó en gran medida en áreas de producción para tareas complejas (mecanizado, soldadura por arco), por su capacidad de moverse suavemente a lo largo de trayectorias continuas (Figura 1.6).



Figura 1. 6 El "Legendary" ABB IRB robot. [16]

En el mismo año, la empresa japonesa Hitachi desarrolló el robot HI-T-HAND Expert, que es relevante en la historia de la robótica industrial por la precisión que alcanzaba en las operaciones de inserción (podía insertar piezas mecánicas con una holgura de unos 10 micrómetros). También estaba provisto de un sistema de control de retroalimentación de fuerza y un mecanismo de muñeca flexible.

Los robots industriales de la tercera generación (convencionalmente comprendidos entre 1978 y 1999) se caracterizaron por un mayor grado de interacción tanto con el operador como con el entorno, a través de algún tipo de interfaces complejas (como la visión o la voz). También tenían algunas capacidades de auto programación, pudiendo reprogramarse, para ejecutar diferentes tareas [15]. Estos robots estaban provistos de servocontroles y podían ejecutar tareas complejas, moviéndose de un punto a otro o por caminos continuos. Se podían programar en línea (el operador podía usar una caja de aprendizaje con un teclado) o fuera de línea, estando conectados a un PLC o una PC, lo que permitía usar un lenguaje de alto nivel para la programación de movimiento y habilitaba los robots para conectarse con un CAD o una base de datos.

La posibilidad de programación fuera de línea de alto nivel amplió el potencial operativo de los robots: por ejemplo, podían elaborar datos a partir de la lectura del sensor, con el fin de ajustar los movimientos del robot teniendo en cuenta los cambios en el entorno (por ejemplo, cambios de posición y orientación de las piezas de trabajo). Además, las capacidades de diagnóstico podían mejorarse en gran medida: estos robots podían producir no solo una indicación de detección de fallas, sino también un informe sobre la ubicación y el tipo de falla. Además, algún tipo de "inteligencia" estaba presente en los robots de la tercera generación, con algunas capacidades adaptativas (aunque limitadas). Estas capacidades podían emplearse en algunas tareas más complejas (como inspección táctil, operaciones de ensamblaje, soldadura por arco), utilizando los datos provenientes de los sistemas de visión o percepción para ubicar los objetos y las piezas de trabajo y

guiar los movimientos articulares de acuerdo con la tarea a realizar, teniendo en cuenta la posibilidad de pequeños cambios en la posición de los objetos.

En 1978, el científico japonés Hiroshi Makino de la Universidad de Yamanashi propuso una estructura cinemática novedosa. Dicha estructura estaba formada por tres articulaciones revolucionarias con ejes paralelos y una articulación prismática al final de la cadena cinemática (Figura 1.7). El robot con esta estructura se denominó SCARA (un acrónimo de “Brazo robótico de ensamblaje de cumplimiento selectivo”), ya que su cumplimiento en la dirección horizontal resultó más bajo que el cumplimiento en la dirección vertical. En robótica, cumplimiento significa que un robot tiene flexibilidad en una o más de sus articulaciones (en ocasiones en su estructura mecánica). Por este motivo, además de por la ligereza de la cadena cinemática (que permitía un controlador más sencillo y rápido), este robot resultaba idóneo para ser empleado en tareas como el montaje de pequeños objetos [17].



Figura 1. 7 Uno de los primeros prototipos de Robot SCARA, diseñado por Hiroshi Makino. [16]

Otra mejora técnica relevante en robótica industrial fue la aparición del robot accionado por actuador directo. El primer prototipo de este tipo fue el CMU Direct Drive Arm [18], desarrollado en 1981 por Kanade y Asaka en la Carnegie Mellon University (Pittsburgh, EE. UU.). Este tipo de robot presentaba una mayor precisión y operaciones más rápidas porque los motores se conectaban

directamente a los brazos eliminando la necesidad de sistemas intermedios de engranajes o cadenas.

Ambos hallazgos antes mencionados se emplearon en el AdeptOne (Figura 1.8), el primer robot SCARA de accionamiento directo disponible comercialmente (1984).



Figura 1. 8 Ejemplos de AdeptOne de SCARA robots. [16]

A pesar de los importantes avances experimentados en la década de 1980, la necesidad de robots que pudieran realizar tareas a gran velocidad impulsó la investigación científica a diseñar estructuras cinemáticas innovadoras. La idea de emplear cadenas cinemáticas paralelas en lugar de las cadenas cinemáticas en serie clásicas propuso y condujo a un tipo de robot ligero con la capacidad de moverse a alta velocidad. El arquetipo de este tipo de robots fue el robot Delta (que apareció en 1992), concebido por el científico suizo Reymond Clavel en la Ecole Poly-Technique Fédérale de Lausanne (EPFL). Este tipo de robot, diseñado por Clavel en su tesis doctoral, tenía tres DOF traslacionales y un DOF rotacional [19]. Con respecto a los robots en serie, los robots paralelos presentaban un espacio de trabajo más pequeño, pero la capacidad de operar a una velocidad mucho mayor. La arquitectura cinemática del robot Delta se copió en muchos manipuladores paralelos, dedicados a operaciones de recogida y colocación de alta velocidad.

La primera aplicación de los robots Delta fue desarrollada por la empresa suiza Demarex en 1992: seis robots Delta operaban dentro de una celda de trabajo para cargar pretzels en bandejas (Figura 1.9). Algunos años después (1998) ABB desarrolló el Flex-Picker, el robot de recolección más rápido del mundo, basado en la estructura del robot Delta (Figura 1.10).



Figura 1. 9 Robots Delta en las celdas de trabajo de Demarex para el empaquetamiento de pretzels. [16]



Figura 1. 10 El robot FlexPicker de ABB. [16]

El final de la tercera generación de la robótica se fija generalmente al final del siglo; se considera que los robots industriales fabricados a partir del año 2000 pertenecen a la cuarta generación (que se extiende hasta los días actuales). Los robots de hoy en día cuentan con capacidades "inteligentes" de alto nivel (como

realizar cálculos avanzados, razonamiento lógico, estrategias complejas y comportamiento colaborativo).

1.2 Antecedentes históricos del control automático

Desde la revolución industrial iniciada en el siglo XVIII, la automatización ha sido la fuerza principal a la hora de intentar racionalizar el proceso de producción. Más tarde, con las primeras computadoras, la integración de circuitos, la ayuda de los sistemas integrados por una central de cómputo y también con robots industriales fue posible automatizar los procesos. Para comprender como todo esto integra lo que es la robótica, se ahondará en conceptos de control automático.

A pesar de que los robots industriales son capaces de realizar correctamente una gran variedad de actividades, a simple vista parecería innecesario desarrollar investigación sobre el tema de control de robots. Sin embargo, es importante resaltar que en aplicaciones específicas es indispensable contar con el modelo dinámico del robot manipulador, así como tener una adecuada estructura matemática de control de alto desempeño práctico para no saturar los actuadores del robot y realizar con exactitud la tarea encomendada al robot. De ahí que el diseño de controladores sigue siendo un área de intensa investigación por parte de universidades, centros de investigación y fabricantes de robots.

El diseño de controladores ofrece grandes retos teóricos que resuelven substancialmente problemas de origen práctico, más aún, su estudio resulta indispensable en aquellas aplicaciones que no pueden realizarse por medio de los robots comerciales. Existen diversas técnicas de diseño de estrategias de control, entre ellas el Molde de Energía la cual es una metodología moderna que se basa en moldear una función de energía potencial dependiente del error de posición e incluye un término con inyección de amortiguamiento o freno mecánico usando la velocidad articular. El objetivo es lograr que $\lim_{t \rightarrow \infty} (\dot{q}^T(t), \check{q}^T(t))^T = 0$, donde la velocidad de movimiento del robot es denotada por el vector \dot{q} , el vector de errores

de posicionamiento es $\tilde{q} = q_d - q$; el algoritmo de control se encarga de llevar en forma asintótica a cero, q_d representa el vector de posiciones deseadas donde el usuario quiere colocar al robot, q es el vector de posición actual del robot, τ es el vector par aplicado y representa la entrada al robot, siendo la energía aplicada a cada una de las articulaciones para que el robot se pueda mover desde su posición inicial a la posición deseada [20]. La figura (Figura 1.11) muestra el diagrama de bloques para un sistema de control de posicionamiento de robots industriales.

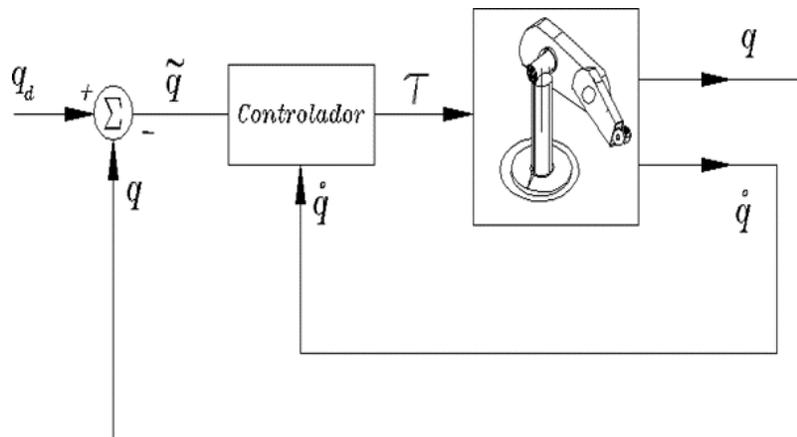


Figura 1. 11 Ejemplo de diagrama de control. Autoría propia.

El control automático de robots manipuladores involucra matemáticas, dinámica no lineal, sensores para retroalimentar la posición de movimiento y técnicas de control de diseño para algoritmos de control. Los conceptos de control clásico Nyquist, Bode, Root Locus son términos específicos de sistemas lineales y por lo tanto no se pueden aplicar a los robots manipuladores, debido a que son sistemas dinámicos no lineales.

Como parte de las técnicas de análisis y diseño de sistemas no lineales en particular se tiene la teoría de estabilidad de Lyapunov, la cual es una herramienta adecuada que permite generar una metodología de diseño que produce una familia extensa de algoritmos de control, en especial moldeo de energía, la cual como mencionamos anteriormente, es una técnica de control moderna que emplea todo el rigor matemático para diseñar una variedad infinita de nuevos esquemas

de control [21]. En general, el desempeño de estos algoritmos de control supera al esquema clásico de control proporcional derivativo PD.

Para que los objetivos de control se cumplan es necesario analizar y estudiar los fenómenos dinámicos que intervienen en el sistema mecánico del robot manipulador, por otro lado, la estructura teórica que explica la física del robot es el modelo dinámico y está formado por una ecuación diferencial no lineal que se obtiene de las leyes de la física usando las ecuaciones de movimiento de Euler-Lagrange [22].

El control de posición de robots manipuladores es el más simple de los problemas de control, al igual que el de trayectoria el cual se aplica cuando el robot se encuentra controlado en posición y velocidad simultáneamente. También se pueden usar sensores externos para realizar control de fuerza, control de impedancia y control mediante información visual (visual servoing).

1.2.1 Mecanización y automatización

Hasta hace medio siglo, la automatización era casi sinónimo de mecanización, el desarrollo de maquinaria y otros dispositivos técnicos sustituyó al trabajo manual. Los inconvenientes son los elevados costes y que los equipos son muy rígidos. Antes de producir un nuevo producto, se tenía que reconstruir toda la línea de producción. Esto resultó en esa mecanización que se aplicó principalmente en industrias que tienen producción en masa, como la industria del automóvil.

Se pueden encontrar muchos ejemplos de automatización durante la industrialización, donde uno de los principales pensamientos fue automatizar el trabajo manual. Desde un punto de vista sueco de Christoffer Polhem el cual era un empresario automotriz. Influenciado por los cambios en los fabricantes de metales y textiles franceses e ingleses, construyó máquinas para cortar ruedas dentadas para relojes que utilizan energía hidroeléctrica en la fábrica de

Stjernerund. La automatización de las ruedas dentadas parecía dudoso al principio, pensaron los relojeros, porque los productos hechos a mano fue una cuestión de orgullo profesional. Polhem también construyó muchas otras máquinas que racionalizó el trabajo manual, como hilanderías y laminadores.

1.3 clasificación de los robots

A continuación, se presenta una clasificación general de robots, se muestran ejemplos de cada una y una breve explicación de su historia y en que consiste ese tipo de robot. Actualmente existe una gran variedad de robots con diversas estructuras geométricas y mecánicas que definen su funcionalidad y aplicación, mostramos la clasificación general en la Tabla 1.1.

Móviles	Humanoides	Industriales
Terrestres: ruedas, patas.	Diseño complejo	<ul style="list-style-type: none"> • Brazos mecánicos • Robots manipuladores

Tabla 1.1 Clasificación de robots

1.3.1 Los Robots Móviles

Los robots móviles pueden ser clasificados de acuerdo con el medio en el que se desplacen: terrestres, marinos y aéreos. Los terrestres generalmente se desplazan mediante ruedas o patas; tienen aplicaciones en rastreo y traslado de objetos, evasión de obstáculos, traslado de instrumental quirúrgico en hospitales, limpieza del área del hogar, ambientes cooperativos y en la industria donde se emplean para análisis e inspección de fisuras en gaseoductos y contenedores de petróleo, por ejemplo. Otra clase de robots móviles como AIBO de Sony son los tipos

mascota (robots perros y gatos) diseñados para ser versátiles en sus movimientos (Figura 1.12).



Figura 1. 12 Robots mascotas en la historia. [25]

Los robots móviles marinos son robots submarinos equipados con sensores especiales para la navegación dentro del agua como son el sonar, radar, visión telescópica, giroscopio, además que poseen sistemas electrónicos especializados que les permiten sumergirse y/o elevarse. Los robots aéreos son aeronaves no tripuladas como helicópteros o pequeños aviones operados a control remoto, aunque hoy en día son más comunes los drones (UAS), los cuales pueden proporcionar imágenes aéreas para reconocimiento de terrenos y evalúan en profundidad la generación de modelos digitales de terreno, ortofotos y planos acotados utilizando técnicas fotogramétricas.

1.3.2 Los robots humanoides

A la ciencia ficción y los medios de comunicación les gusta representar a los robots en forma humanoide. Todos estamos familiarizados con R2-D2 y 3-CPO, los personajes robóticos de las películas de Star Wars, pero el concepto se remonta a mucho tiempo atrás. En el siglo XVIII, un grupo de relojeros suizos, Pierre y Henri-Louis Jaquet-Droz y Jean-Frédéric Leschot, construyeron autómatas humanoides para demostrar sus habilidades mecánicas y publicitar sus relojes. Hoy en día, muchas empresas construyen robots humanoides por razones similares.

Los robots humanoides son una forma de robot móvil autónomo con un diseño mecánico extremadamente complejo para mover los brazos y para la locomoción de las piernas. Los robots humanoides se utilizan para investigar la mecánica del caminar y la interacción hombre-máquina. Se han propuesto robots humanoides para realizar servicios y mantenimiento en una casa o una estación espacial. Se están considerando para brindar atención a los ancianos que podrían sentirse ansiosos en presencia de una máquina que no parecía humana. Por otro lado, los robots que se parecen mucho a los humanos pueden generar repulsión, un fenómeno conocido como el valle inquietante.

Los robots humanoides pueden ser muy difíciles de diseñar y controlar. Son costosos de construir con múltiples articulaciones que se pueden mover de muchas formas diferentes. Los robots que usan ruedas u orugas son los preferidos para la mayoría de las aplicaciones porque son más simples, menos costosos y robustos (Figura 1.13).



Figura 1. 13 Ejemplos de robots humanoides en el siglo XXI. [26]

1.3.3 Los robots industriales

Los robots industriales son el tipo de robots más populares, debido a la importancia que ocupan en el sector industrial como herramientas clave para la modernización de las empresas. Hoy en día la automatización de procesos industriales es realizada a través de robots y esto trae como consecuencia competitividad, productividad, eficiencia y rentabilidad de las empresas.

Los primeros robots fueron robots industriales que reemplazaron a los trabajadores humanos que realizaban tareas simples y repetitivas. Las líneas de montaje de la fábrica pueden operar sin la presencia de humanos, en un entorno bien definido donde el robot tiene que realizar tareas en un orden específico, actuando sobre objetos colocados con precisión frente a él (Figura 1.14).



Figura 1. 14 Robots en línea de ensamblaje de fábrica de autos por Alejandro Serrano.

Se podría argumentar que estos son realmente autómatas y no robots. Sin embargo, los autómatas actuales a menudo se basan en sensores en la medida en que pueden considerarse robots. Sin embargo, su diseño se simplifica porque funcionan en un entorno personalizado al que los humanos no pueden acceder mientras el robot está funcionando.

Sin embargo, los robots de hoy necesitan más flexibilidad, por ejemplo, la capacidad de manipular objetos en diferentes orientaciones o de reconocer diferentes objetos que deben empaquetarse en el orden correcto. El robot puede ser necesario para transportar mercancías desde y hacia los almacenes. Esto aporta autonomía adicional, pero la característica básica permanece: el entorno está más o menos restringido y se puede adaptar al robot.

Se requiere flexibilidad adicional cuando los robots industriales interactúan con humanos y esto introduce fuertes requisitos de seguridad, tanto para brazos robóticos como para robots móviles. En particular, la velocidad del robot debe reducirse y el diseño mecánico debe garantizar que las piezas móviles no sean un peligro para el usuario. La ventaja de los humanos que trabajan con robots es que cada uno puede realizar lo que mejor sabe hacer: los robots realizan tareas repetitivas o peligrosas, mientras que los humanos realizan pasos más complejos y definen las tareas generales del robot, ya que reconocen rápidamente errores y oportunidades para mejoramiento.

1.4 Aplicaciones industriales

La Robótica Industrial, es la mejor alternativa para aumentar la productividad con la agilización de procesos de producción y ahorrar en gastos por sueldos. La Industria Manufacturera es una de las más representativas, al tener empleados perfectos capaces de ejercer tareas como transporte de carga pesada, material radiactivo, empaquetamiento, etc. Grandes empresas internacionales cuentan actualmente con alta tecnología robótica, debido a la importancia que brindaron a la investigación y desarrollo tecnológico, por ejemplo, los gigantes asiáticos son los que tienen el mayor margen de producción debido a la aplicación de procesos automatizados. A continuación, se muestra en la Tabla 1.2 un par de aplicaciones generales más populares en la industria.

Manipulación:	Procesado:
<ul style="list-style-type: none"> • Ensamblado, desensamblado • Paletizado • Empaquetado 	<ul style="list-style-type: none"> • Soldadura • Pintura • Mecanizado

Tabla 1.2 Aplicaciones de robots industriales

1.4.1 Aplicaciones de manipulación

La automatización robótica ha permeado en numerosas áreas de la industria realizando tareas como soldadura, montaje, envío, manipulación de materias primas y empaque de productos. Es por lo que cada vez más fabricantes están aprovechando la automatización robótica para una mayor cantidad de aplicaciones, en el caso particular de las aplicaciones de los robots manipuladores son todas aquellas en las que la función del robot es transferir o mover piezas, materiales o herramientas de un lugar a otro, estas aplicaciones incluyen:

- **Paletización de cajas:** Un robot móvil es capaz de desplazarse por el almacén y recoger las unidades y referencias que incorporan un pedido concreto para un cliente.
- **Colocación de las baterías en los vehículos:** En el proceso de montaje de un coche, un robot antropomórfico recoge la batería de una pila de pallets y la coloca en el interior del motor del coche.
- **Empaquetado de producto:** La visión es de uso recurrente en el empaquetado correcto de distintos productos genéricos y de alimentación.
- **Procesos de taladrado, remachado y atornillado de alta precisión:** En la industria aeronáutica se utilizan cabezales de robots multifuncionales para el remachado en el fuselaje de un avión.

Una característica clave en los robots manipuladores es que la capacidad de realizar la tarea asignada de forma exitosa depende grandemente de la pinza (gripper), esta es una herramienta conformada por uno o varios actuadores, esta

herramienta se diseña para la tarea particular que hará cada robot, a continuación, se ahonda más en el tema:

Pinzas y dedos: En el mercado se pueden encontrar pinzas con distintos números de piezas articuladas que simulan dedos y a la vez una mano, la más común son de dos dedos la cual conforma una pinza, el número de dedos a utilizar depende mucho de la forma geométrica de la pieza a manejar, por ejemplo, las pinzas con tres dedos articulados son comúnmente usadas para sujetar piezas esféricas y cilíndricas.

Herramientas: A la muñeca del robot industrial no solo se le puede acoplar un manipulador, sino también herramientas específicas, como atornilladores, soldadores, láser, dosificadores de adhesivo, dosificadores de pintura, pulidoras, etc.

Gripper versátiles: Finalmente muchos procesos industriales son específicos y distintos para cada método de fabricación. Por ello la solución final requiere del diseño específico y a medida de útiles, garras o manos especiales que no se encuentran en el mercado.

Gripper según su accionamiento: De cierre angular o paralelo: Las pinzas de robot estándar tienen dos tipos de movimientos de cierre, paralelo o angular, y el movimiento puede ser operado mediante hidráulica, neumática, eléctrica o por resortes.

De vacío: Eficaz cuando el robot (colaborativo o industrial) se encarga de la manipulación de chapas, madera, plástico, barriles, vidrios, lunas de cristal y alimentos, entre otras aplicaciones. Se prefiere el uso de vacío para piezas no magnéticas, frágiles, delicadas y ligeras.

Dependiendo del material con el que se vaya a trabajar es importante el uso de ventosas específicas para cada ocasión. La configuración de los labios de la ventosa, su forma, tamaño y la rigidez del fuelle se tienen en cuenta según la dureza del material y las características superficiales del mismo. Por ejemplo, las ventosas elásticas se utilizan para piezas rígidas y las rígidas para piezas elásticas.

Cabe destacar que el robot manipulador a rehabilitar Pegasus; es empleado en aplicaciones de manipulación como lo es: ensamble, empaquetado, Pick & place, palatización, entre otras. La Figura 1.15 ilustra el brazo manipulador Pegasus de AMATROL. En la Figura 1.16 se ilustra uno de los robots más populares para las tareas de pick & place fabricado por la compañía Asea Brown Boveri (ABB).



Figura 1. 15 Robot manipulador Pegasus de AMATROL.



Figura 1. 16 Robot IRB 910SC de ABB tipo SCARA.

1.4.2 Aplicaciones por proceso

Podemos decir que la robotización de procesos tiene dos grandes objetivos, por un lado, la reducción de costes, y por el otro, un incremento de la producción y la calidad. A continuación, se presentan más a detalle algunas de las tareas más comunes que son realizadas por robots manipuladores de proceso:

Soldadura de puntos: En este caso la herramienta usada al extremo del brazo es un gripper de soldar por puntos. El robot se programa generalmente por medio de una "caja de enseñanza" (teach box) para que suelde siguiendo un mapa determinado de coordenadas y actuando la pinza en cada punto (Figura 1.17). Las partes para soldar deben estar en la posición correcta o el robot soldará en puntos desplazados. La caja de enseñanza tiene un teclado con el cual pueden controlarse las articulaciones y actuar de la herramienta de trabajo en la secuencia deseada por el operador. Las señales producidas por el operador se van guardando en la memoria del microcomputador controlador del brazo y pueden ser después repetidas un número ilimitado de veces.

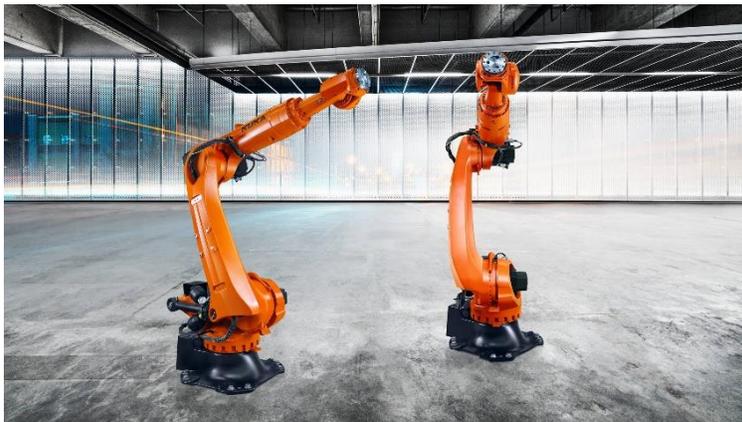


Figura 1. 17 Robot KR Quantec para soldadura por puntos de KUKA.

Pintura "espray": La herramienta de trabajo es una pistola de pintar. El movimiento no es de punto a punto, sino continuo. En este caso es más práctico tener una réplica de peso ligero y tamaño natural del robot, con los mismos sensores que este (Figura 1.18). Un operario especializado mueve la pistola como

si estuviera pintando y los movimientos son capturados, a través de los sensores, a la memoria del microprocesador. Al terminar la tarea, el robot real se conecta al microprocesador de control y los movimientos son replicados fielmente. Debido al método de programación descrito, se usa el término "leading by the nose" en inglés, o sea "guiar por la nariz".



Figura 1. 18 Robot EcoRP E043i para pintura automotriz de Dürr.

Corte por láser: Esta tarea se lleva a cabo por robots diseñados para trabajar a alta velocidad y a la vez prestando alta rigidez y precisión para proporcionar un seguimiento de trayectoria preciso en los cortes, por lo general el gripper con el que están equipados pueden realizar diferentes tareas con una modificación pequeña, como lo son la soldadura laser, corte por plasma y aplicaciones de dispensación (Figura 1.19).



Figura 1. 19 Robot articulado MC2000 II de YASKAWA.

1.5 Robótica pedagógica en el aprendizaje de ingeniería

En los últimos años, la robótica en la educación ha surgido como una actividad complementaria a los planes educativos, esta es interdisciplinaria y se basa en proyectos que ofrecen nuevos e importantes beneficios para la educación en todos los niveles. Se ha convertido en una necesidad esencial para el desarrollo de la mente de los estudiantes y su preparación para el mundo laboral de la nueva era.

Con la robótica pedagógica se busca mejorar en gran manera el enfoque del aprendizaje tradicional, en la actualidad aún hay maestros que dictan todo lo que él o ella saben acerca de la asignatura a un grupo de estudiantes y estos anotan toda la información, la cual finalmente solo es retenida en la memoria por un corto periodo de tiempo hasta el día del examen. Incluso cuando se les pide a los estudiantes aplicar la información recibida a una supuesta situación real, utilizan esta información de manera sistemática, no razonan lo aprendido y muchas veces la situación planteada por el maestro es un problema arbitrario sin importancia para el estudiante. Por lo tanto, si el maestro aborda la experiencia del aula con escenarios reales, practicas con objetos, herramientas, recursos de la vida real y proyectos que despierten el interés de estos por aprender, logrará desarrollar más habilidades y aptitudes para el desarrollo profesional en el estudiante [23].

Muchas escuelas tienen computadoras, estas se utilizan frecuentemente como herramientas para ayudar a los estudiantes a practicar en un formato de simulacro lo que han estado aprendiendo del plan de estudios del aula. No están allí para permitir el descubrimiento de las áreas de interés de los estudiantes. Existe la creencia generalizada de que, en el aprendizaje, los robots pueden ser un medio eficaz para promover el compromiso, motivar y desarrollar conjuntos de habilidades prácticas, más allá del enfoque de la robótica en sí. La robótica es un área multidisciplinaria apasionante que va a dominar el siglo XXI. La robótica en la industria está entrando en un nuevo período de rápido crecimiento [24].

Los robots pueden verse como herramientas informáticas con ambas habilidades; computacionales y mecánicas para realizar tareas orientadas al movimiento físico. Los robots permiten desarrollar habilidades de electrónica tecnología y proporcionar a los estudiantes oportunidades para el aprendizaje basado en proyectos. En el contexto del e-learning, los robots son cada vez más vistos como un medio para promover la participación, el interés y diversión en el aprendizaje [25].

1.5.1 ¿Qué es un robot pedagógico?

Para distinguir entre robots educativos, robots industriales y comerciales, cabe señalar que ambos tipos abarcan las mismas características generales. Sin embargo, lo que distingue a los robots pedagógicos es que son más seguros para su uso por estudiantes y que son de menor costo material. Además de eso, contienen instrucciones que guían a los estudiantes sobre cómo programarlos e interactuar con sus componentes como, sus sensores, motores y grippers.

Estos Robots Pedagógicos reciben órdenes a través de una computadora provista de un programa, además de algunas aplicaciones que permiten al alumno realizar prácticas y experimentos en realidad simulada. Este tipo de robots se pueden aprovechar para desarrollar conceptos y habilidades de programación en los estudiantes, este método educativo transfiere al estudiante de lo teórico a la aplicación práctica, a través de la integración de las diferentes asignaturas educativas de ciencia, tecnología, ingeniería y matemáticas.

El papel del robot pedagógico no se limita solo al aspecto del intercambio tecnológico, sino que también contribuye al refinamiento de la preparación profesional del estudiante acorde con las exigencias de la época y las habilidades requeridas en el siglo XXI.

Los componentes esenciales de un robot pedagógico excluyendo las partes mecánicas son:

- Kits Arduino

- DC-Servomotores
- Sensores (encoders/ultrasónicos/luces/compás)
- Llantas (omni/convencional)
- Software para el control del sistema

1.5.2 Importancia y beneficios de usar la robótica en la docencia educativa

A continuación, cito algunos puntos de la importancia y beneficios de la robótica en base al extracto de la conferencia de Robots Pedagógicos [26].

- Ayuda a identificar las diferentes funciones de la mecánica, a desarrollar la habilidad de diseñar piezas en software CAD, a ensamblar y enlazar.
- Permite al alumno adentrarse en las características físicas y mecánicas de los materiales, en el área de programación y control, así como en el desarrollo de los algoritmos especiales para robots, lo que permite controlar los sensores que se relacionan con el entorno (motores, sensores ultrasónicos y de visión, actuadores).
- Desempeña un papel importante en el nivel social del estudiante, ya que fomenta el espíritu del trabajo en equipo, la responsabilidad y manejar el intercambio de roles, además de la habilidad de suministrar y acelerar las tomas de decisiones con las condiciones presentes durante y después de los proyectos.
- Les da a los estudiantes la oportunidad de escoger el papel a desempeñar en la elaboración de proyectos.
- Fomenta y desarrolla las habilidades para elaborar trabajos manuales.
- Fomenta la estrategia de aprendizaje basada en proyectos y metas.
- Desarrolla y mejora las habilidades creativas, de pensamiento, de razonamiento y de resolución de problemas.
- Vincula lo aprendido con la vida práctica.

- Ayuda a establecer el concepto de que el estudio no siempre es sistemático y poco atractivo.

1.5.3 Robots pedagógicos preensamblados y kits.

Muchos robots educativos están diseñados como robots móviles preensamblados. La Figura 1.20 muestra el robot Thymio de Mobsya y la Figura 1.21 muestra el robot Dash de Wonder Workshop. Estos robots son relativamente económicos, robustos y contienen una gran cantidad de sensores y componentes de salida como luces. Una ventaja importante de estos robots es que puede implementar algoritmos robóticos "listos para usar", sin invertir horas en diseño y construcción mecánicos. Sin embargo, los robots preensamblados no se pueden modificar, aunque muchos admiten extensiones de construcción utilizando, por ejemplo, LEGO.



Figura1. 20 Robot educacional Thymio de Mobsya. Figura1. 21 Robot educacional Dash de Wonder Workshop.

Los kits de robótica LEGO® Mindstorms (Figura 1.22) se introdujeron en 1998.1 Un kit consta de LEGO® estándar ladrillos y otros componentes del edificio, junto con motores y sensores, y un ladrillo programable que contiene la computadora que controla los componentes del robot. La ventaja de los kits de robótica es que son flexibles: puedes diseñar y construir un robot para realizar una tarea específica, limitada solo por tu imaginación. También se puede utilizar un kit de robótica para enseñar a los estudiantes el diseño mecánico. Las desventajas de los kits de robótica es que son más caros que los simples robots preensamblados

y que la exploración de los algoritmos de robótica depende de la capacidad de cada uno para implementar con éxito un diseño mecánico robusto.

Una tendencia reciente es reemplazar las colecciones fijas de ladrillos por piezas construidas con impresoras 3D. Un ejemplo es el brazo robótico Poppy Ergo Jr (Figura 1.23). El uso de piezas impresas en 3D permite una mayor flexibilidad en la creación de la estructura mecánica y una mayor robustez, pero requiere acceso a una impresora 3D.



Figura1. 22 Kit educacional LEGO Mindstorms Figura1. 23 Kit de brazo robotico Poppy Ergo Jr

1.6 Sistema operativo robótico (ROS)

Robot Operating System (ROS) es un middleware robótico, es decir, una colección de frameworks para el desarrollo de software de robots. ROS se desarrolló originariamente en 2007 bajo el nombre de switchyard por el Laboratorio de Inteligencia Artificial de Stanford para dar soporte al proyecto del Robot con Inteligencia Artificial de Stanford (STAIR2). Desde 2008, el desarrollo continuó principalmente en Willow Garage, un instituto de investigación robótico con más de veinte instituciones colaborando en un modelo de desarrollo federado.

A pesar de no ser un sistema operativo, ROS provee los servicios estándar de uno de estos tales como la abstracción del hardware, el control de dispositivos de bajo

nivel, la implementación de funcionalidad de uso común, el paso de mensajes entre procesos y el mantenimiento de paquetes. Está basado en una arquitectura de grafos donde el procesamiento toma lugar en los nodos que pueden recibir, mandar y multiplexar mensajes de sensores, control, estados, planificaciones y actuadores, entre otros. La librería está orientada para un sistema UNIX (Ubuntu - Linux-) aunque también se está adaptando a otros sistemas operativos como Fedora, Mac OS X, Arch, Gentoo, OpenSUSE, Slackware, Debian o Microsoft Windows, considerados hoy en día como 'experimentales' [27].

ROS tiene dos partes básicas: la parte del sistema operativo, `ros`, y `ros-pkg`. Esta última consiste en una suite de paquetes aportados por la contribución de usuarios (organizados en conjuntos llamados pilas o en inglés `stacks`) que implementan las funcionalidades tales como localización y mapeo simultáneo, planificación, percepción, simulación, etc. ROS es software libre bajo términos de licencia BSD. Esta licencia permite libertad para uso comercial e investigador. Las contribuciones de los paquetes en `ros-pkg` están bajo una gran variedad de licencias diferentes.

Entre las versiones actuales y las que están en desarrollo se presentan las siguientes aplicaciones de procesos de ROS:

- Percepción
- Identificación de Objetos
- Segmentación y reconocimiento
- Reconocimiento facial
- Reconocimiento de gestos
- Seguimiento de objetos
- Egomoción
- Comprensión de movimiento
- Estructura de movimientos (SFM)
- Visión estéreo: percepción de profundidad mediante el uso de dos cámaras

- Movimientos
- Robots móviles
- Control
- Planificación
- Agarre de objetos

Entre algunos de los robots que ya utilizan ROS se pueden encontrar:

- Baxter de Rethink Robotics, Inc.
- Robot de Shadow: mano robótica diestra motorizada desarrollada por la empresa Shadow y la cual se está desarrollando mediante el consorcio de un proyecto europeo dentro del marco europeo. Entre los participantes de este proyecto se pueden encontrar la empresa Shadow Robot, la Université Pierre et Marie Curie-Paris (Francia) o la Universidad Carlos III de Madrid (España) [28].
- HERB: desarrollado en CMU dentro del programa de robótica personal de Intel.

1.6.1 Sistema de Archivos

El sistema de archivos está compuesto por los recursos que se encuentran en el mismo programa como:

- Paquetes: los paquetes son la unidad principal para organizar software en ROS. Un paquete puede contener procesos ejecutables (nodos), una biblioteca dependiente, conjuntos de datos, archivos de configuración, o cualquier otra cosa que sea útil para una organización conjunta.
- Manifiestos: proporcionan metadatos sobre un paquete, incluyendo su información de licencia y dependencias, así como información específica del compilador.
- Pilas: es una colección de paquetes que tiene una misma función.

- Manifiestos de pilas: proporcionan datos sobre una pila, incluyendo su información de licencia y sus dependencias en otras pilas.
- Mensajes: definen las estructuras de datos para los mensajes enviados en ROS.
- Servicios: definen la solicitud y estructuras de datos de respuesta de los servicios requeridos por ROS.

1.6.2 Computación a nivel gráfico

La computación a nivel gráfico es la red de ROS que se encarga de procesar todos los datos. Los conceptos básicos son nodos, maestro, mensajes y tópicos, los cuales proporcionan los datos de diferentes maneras:

- Nodos: los nodos son procesos que llevan a cabo cálculos, Ros está diseñado para ser modular en una escala básica. Un sistema de control de robot comprenderá usualmente muchos nodos. Por ejemplo, un nodo controla un teledetector láser, un nodo controla los motores de las ruedas, un nodo realiza la localización, un nodo realiza la planificación de la ruta y otro nodo proporciona una vista gráfica del sistema, y así sucesivamente.
- Maestro: el maestro proporciona registro de nombres y la búsqueda para el resto de la Computación Gráfica. Sin el Maestro, los nodos no serían capaces de encontrar mensajes entre sí, intercambiar o invocar a los servicios.
- Mensajes: los nodos se comunican entre sí pasando mensajes. Un mensaje es simplemente una estructura de datos que comprende los tipos de campos. Los mensajes pueden incluir estructuras arbitrariamente anidadas y matrices (al igual que las estructuras de C).
- Tópicos: Los mensajes se enrutan a través de un sistema de transporte de publicación / suscripción semántica. Un nodo envía un mensaje por publicar a un determinado tópico. El tópico es un nombre que se utiliza para identificar el contenido del mensaje. Un nodo que está interesado en un determinado tipo de datos se suscribe al tópico correspondiente. Puede

haber varios editores y suscriptores concurrentes a un mismo tema, y un único nodo puede publicar y / o suscribirse a múltiples tópicos. En general, los editores y suscriptores no son conscientes de la existencia de lo demás. Se puede pensar en un tópico como un Bus de mensajes. Cada Bus tiene un nombre, y cualquier persona puede conectarse al Bus para enviar o recibir mensajes, siempre y cuando sean del tipo correcto.

1.6.3 Distribuciones de ROS

Una distribución de ROS es un conjunto versionado de librerías, las cuales permiten a los desarrolladores trabajar con una versión estable de código. Cuando estas distribuciones se publican, los cambios se limitan a correcciones de fallos y modificaciones que no alteren el funcionamiento de los módulos principales de la distribución. A continuación, se presentan las distribuciones más importantes y recomendadas respecto a ROS al año 2021:

- **ROS Lunar:**
 - Fecha de publicación: 23/05/2017
 - Plataformas:
 - Ubuntu: Willy; Xenial
 - Debian: Stretch
 - Documentación: <http://wiki.ros.org/lunar/Migration>

- **ROS Kinetic**
 - Fecha de publicación: 23/05/2016
 - Plataformas:
 - Ubuntu: Willy; Xenial
 - Debian: Jessie
 - OS X (Homebrew)
 - Gentoo
 - OpenEmbedded/Yocto

- Documentación: <http://wiki.ros.org/kinetic/Migration>

- **ROS Jade**
 - Fecha de publicación: 23/05/2015
 - Plataformas:
 - Ubuntu
 - OS X (Homebrew)
 - Gentoo
 - Android (NDK)
 - Documentación: <http://wiki.ros.org/jade/Migration>

- **Ros Indigo**
 - Fecha de publicación: 23/05/2014
 - Plataformas:
 - Ubuntu
 - Debian: Wheezy
 - OS X (Homebrew)
 - Gentoo
 - OpenEmbedded/Yocto
 - Android (NDK)
 - Documentación: <http://wiki.ros.org/indigo/Migration>

En este proyecto se decidió utilizar la distribución de ROS Kinetic ya que esta distribución está diseñada para controlar los movimientos cinéticos de sistemas robóticos, a la vez que es compatible con la herramienta MoveIt! la cual está diseñada específicamente para brazos manipuladores. Con esta configuración del sistema podemos generar un control robusto sobre el cual basarnos y desarrollar el proyecto de restauración del robot Pegasus.

Aunado a la robustez de la distribución, tenemos una gran variedad de sensores soportados, de diferentes marcas y funcionalidades, y todos ellos con un apartado

específico en la página web del sistema, en el caso de que el sensor a utilizar no sea soportado, se tiene la opción de generar una documentación propia necesaria para su uso, a continuación un listado de algunos sensores soportados:

- Sensores de 1 dimensión
 - Sharp IR range finder (via ArbotiX RoboController)
- Sensores 2D
 - SICK LMS2xx lasers
 - SICK LMS1xx lasers
 - Sick(R) S300 Professional
 - Leuze rotoScan laser rangefinder driver (ROD-4, RS4)
 - Neato XV-11 Laser Driver
 - Sick LD-MRS Laser Driver
 - SICK TiM3xx lasers
- Sensores 3D
 - Mesa Imaging SwissRanger devices (3000/4000)
 - OpenNI driver for Kinect and PrimeSense 3D sensors
 - Velodyne HDL-64E 3D LIDAR
 - Forecast 3D Laser with SICK LIDAR

1.6.4 ROS y la docencia

El sistema ROS como se ha mencionado es un sistema que permite construir un sistema robusto y funcional de manera sencilla, este sistema maneja desde la robótica más simple hasta el apartado más complejo, tiene soporte para la mayoría de los robots que se usan en la industria y en la actualidad este sistema se usa en los centros de enseñanza e investigación.

Al mes de noviembre de 2021, hay más de 156 instituciones docentes de todo el mundo trabajando e investigando con el sistema operativo de ROS. A nivel global encontramos una gran variedad de instituciones docentes como: Stanford, Instituto

Tecnológico de Massachusetts (MIT) y la universidad de estudios aeroespaciales de Toronto entre otras.

En la siguiente ilustración podemos ver un mapa proporcionado por ROS donde están señaladas las ubicaciones de escuelas, investigadores, usuarios comunes de ROS que han creado o usado un repositorio de ROS.



Figura 1. 24 Mapa de usuarios de ROS en el mundo [29].

CAPITULO II. GENERALIDADES

2.1 Introducción

En este capítulo abordaremos los detalles del problema a atacar que es la rehabilitación del robot Pegasus de 5 grados de libertad (GDL), se detallan algunos conceptos que son necesarios conocer para llevar a cabo el proyecto.

Los conceptos que son presentados abordan desde los tipos y estructuras de los subsistemas que conforman la morfología del robot Pegasus, continuando con una introducción a las configuraciones de los robots manipuladores, otro aspecto importante que se abarca son los motores y sensores de seguridad; así mismo se mencionan algunos tipos de control.

2.2 Morfología de los robots

Como morfología se denomina la rama de una disciplina que se ocupa del estudio y la descripción de las formas externas de un objeto en este caso se enfoca a las estructuras de los robots. La palabra se compone de los vocablos griegos μορφή (morphé), que significa 'forma', y λόγος (lógos), 'tratado' [30]. En este caso se enfoca a las estructuras de los robots. Para ello se definirá que es un grado de libertad, clasificar por movimiento los tipos de articulaciones que pudieran contener los robots.

2.2.1 Grado de libertad

El grado de libertad (GDL) de un sistema es el número de parámetros independientes que se necesitan para definir unívocamente la posición de un sistema mecánico en el espacio en cualquier instante. De forma similar, diremos que el mínimo número de coordenadas necesarias para especificar de forma única

las posiciones de todos los componentes de un sistema de cuerpos rígidos, será el número de grados de libertad del sistema. Cada movimiento independiente que puede realizar una articulación con respecto al movimiento anterior es definido como un grado de libertad y es además el número de parámetros independientes que fijan la situación (posición y orientación) del elemento terminal [31].

La estructura de un robot se puede dividir en pequeños elementos; barras lineales, en ángulo, curvas y estos elementos están unidos por nudos. Las direcciones en las que cada nudo es capaz de rotar y girar son conocidas como grados de libertad del nudo. En el caso más general en el espacio tridimensional un nudo es capaz de rotar en tres direcciones y también de desplazarse en tres direcciones. Entonces se tendrán un total de seis grados de libertad para el nudo (tres rotaciones y tres desplazamientos rectilíneos) como se ilustra en la Figura 2.1.

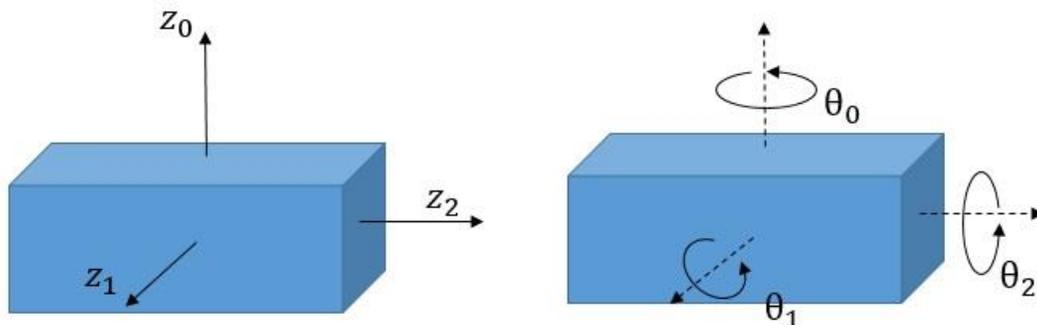


Figura 2. 1 Ilustración de grados de libertad de un nudo. [35]

2.2.2 Tipos de articulaciones

Los manipuladores industriales están básicamente constituidos por elementos rígidos (también llamados eslabones), los cuales son conectados por uniones (también llamados articulaciones) dentro de una cadena cinemática abierta. Estas articulaciones son usualmente de dos tipos: rotación o prismáticas (también llamadas lineales). Estos dos tipos de articulaciones nos permiten realizar un movimiento relativo de rotación o también un movimiento lineal entre dos cuerpos

rígidos (eslabones). Las articulaciones lineales implican un movimiento deslizante o de translación de las uniones de conexión, este movimiento puede ser generado por pistones o por medio de hacer deslizar el elemento sobre un carril o guía usando dispositivos mecánicos, eléctricos o neumáticos.

Las articulaciones de rotación cuentan con un grado de movilidad que permite un movimiento relativo de rotación alrededor de un eje de rotación común, normalmente consistente en una junta giratoria manejada por motores eléctricos y transmisiones, o por cilindros hidráulicos y palancas.

Se pueden distinguir tres tipos:

- Articulación Rotacional, el eje de rotación es perpendicular a los ejes de las dos uniones (Figura 2.2 a).
- Articulación Cilíndrica, esta realiza un movimiento compuesto, tiene un GDL de articulación lineal y otro de rotación sobre el mismo eje (Figura 2.2 b).
- Articulación Prismática, esta articulación solo tiene 1 GDL, se mueve en línea recta sobre un eje (Figura 2.2 c).

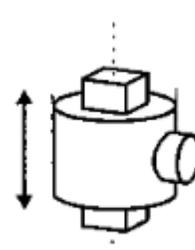
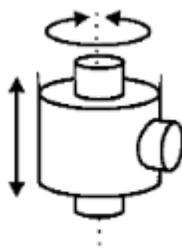
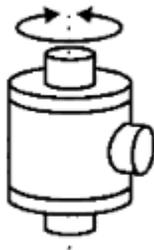


Figura 2. 2 a) Articulación Rotacional (1 GDL) b) Articulación Cilíndrica (2 GDL) c) Articulación Prismática (1 GDL)

También existen otros tipos de articulaciones en la robótica como la esférica, caracterizada por combinar tres giros en tres direcciones perpendiculares en el espacio (Figura 2.3 a), la planar se caracteriza por el movimiento de desplazamiento en un plano, existiendo 2 grados de libertad. (Figura 2.3 b), en el tornillo el grado de libertad consiste en la translación a lo largo de un eje roscado. (Figura 2.3 c):

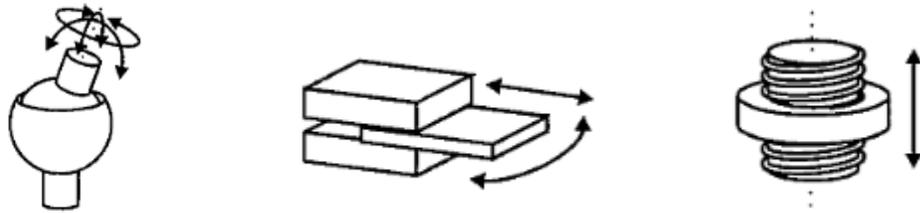


Figura 2.3 a) Articulación Esférica (3 GDL) b) Articulación Planar (2GDL) c) Tornillo (1GDL)

2.2.3 Estructuras de robots manipuladores

La estructura física de la mayoría de los robots industriales tiene mucha similitud con la anatomía de las extremidades superiores del cuerpo humano, por lo que, en ocasiones, para hacer referencia a los distintos elementos que componen el robot, se usan términos como cintura, hombro, brazo, codo, muñeca, etc. Figura 2.4.

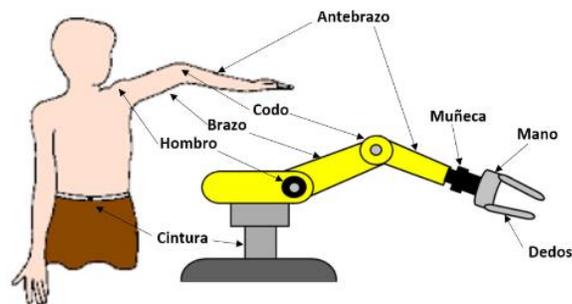


Figura 2.4 Ejemplo de estructura de robot manipulador. [36]

Por lo general la mayoría de los robots manipuladores son cadenas cinemáticas abiertas con las articulaciones de tipo rotación o prismática (con un solo GDL, grado de libertad, cada una), siendo por lo general sencillo encontrar el número de GDL, pues coincide con el número de articulaciones de las que se compone. Los robots con cadena cinemática cerrada y en particular los robots denominados de «estructura paralela», son menos frecuentes, si bien en los últimos años algunos fabricantes de robots ofrecen productos con esta característica. El empleo de diferentes combinaciones de articulaciones en un robot da lugar a diferentes configuraciones, con características para tener en cuenta tanto en el diseño y construcción del robot como en su aplicación.

2.2.4 Estructuras básicas

Al hablar de las estructuras básicas más utilizadas en los robots, se debe tomar en cuenta las ventajas como accesibilidad a posiciones, velocidad de desplazamiento y propiedades en cuanto al volumen de trabajo. El espacio de trabajo es una característica fundamental para tomar en cuenta al momento de diseñar la estructura del robot y para tener sus aplicaciones adecuadamente justificadas.

El volumen de trabajo de un robot manipulador es aquel que, el efector final puede alcanzar todos los puntos en base al movimiento dentro de él. El volumen de trabajo es definido por la geometría del manipulador, así como la configuración de sus articulaciones (Figura 2.5).

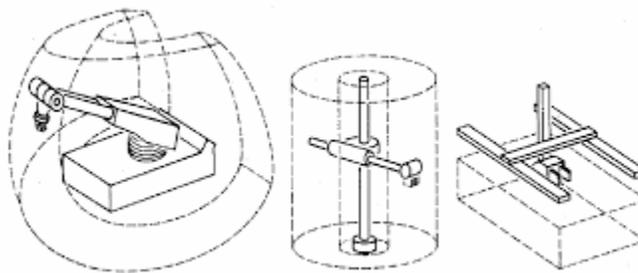


Figura 2. 5 Volumen de trabajo de robots manipuladores. [22]

2.2.5 Configuración Cartesiana

Esta configuración utiliza tres ejes de movimiento lineal (Articulación prismática o lineal) perpendiculares entre sí (Figura 2.6). Esta configuración da lugar a robots que pueden desempeñar las tareas con alta precisión, velocidad y capacidad de carga constante en todo su alcance, estos también cuentan con amplia zona de trabajo y un sistema de control relativamente sencillo.

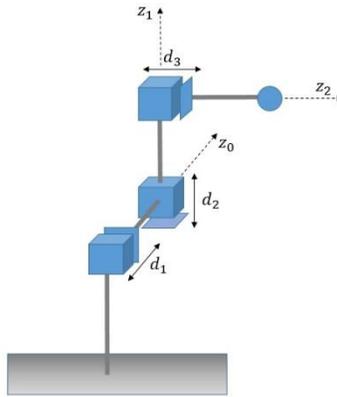


Figura 2. 6 Estructura de robot tipo cartesiano. [35]

En teoría, este tipo de robots también son más fáciles de programar. Sin embargo, hoy en día este argumento podría haber perdido fuerza, ya que comúnmente, los fabricantes de robots tienen su propio software para sus productos y toda la computación y el control se realizan "bajo el capó".

Se aprecia en la Figura (2.7) un robot cartesiano de marca EPSON, este tiene una estructura rígida que se desplaza sobre 3 ejes (X, Y, Z) y cuenta con una alta precisión de repetitividad de $\pm 0.01\text{mm}$. En comparación con las configuraciones que incluyen articulaciones de rotación, presentan una mala relación entre su volumen de trabajo y el espacio que ocupan en planta (Figura 2.8).



Figura 2. 7 Robot cartesiano de 3 ejes EPSON's RP-HMSz

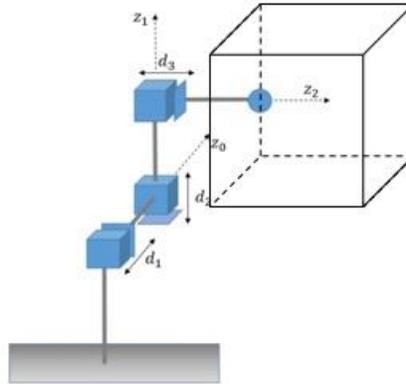


Figura 2. 8 Volumen de trabajo de robot tipo cartesiano. [35]

2.2.6 Configuración Cilíndrica

Este robot manipulador es concebido cuando se tiene una articulación de rotación y dos prismáticas, los puntos que puede alcanzar pueden ser debidamente especificados con coordenadas cilíndricas. Un robot con este tipo de brazo se denomina robot cilíndrico, generalmente los robots de este tipo pueden moverse a lo largo de los ejes (Y, Z) y rotar a lo largo del eje Z. Puesto que las coordenadas del brazo pueden asumir cualquiera de los valores entre los límites superior e inferior especificados, su efector final puede moverse en un volumen limitado, que es una sección de corte dentro del espacio entre los dos cilindros concéntricos (Figura 2.9).

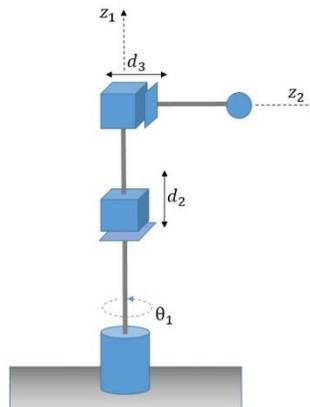


Figura 2. 9 Robot de configuración tipo cilíndrico. [35]

El conjunto de aplicaciones más habituales en las que se utilizan es el tipo de aplicaciones donde se requiere el trabajo de un desplazamiento envolvente cilíndrico combinado con la orientación horizontal de la herramienta se puede apreciar en el volumen de trabajo (Figura 2.10). Por ejemplo, tareas específicas de manipulación y montaje o soldadura por puntos.

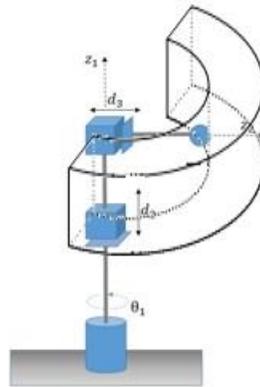


Figura 2. 10 Volumen de trabajo de robot tipo cilíndrico. [35]

En la Figura 2.11 se puede observar un robot un poco “antiguo”. Se fabricó a principios de los años 2000. Es un típico robot manipulador de coordenadas cilíndricas que puede moverse a lo largo de dos ejes y girar alrededor del eje Z, colocando así su herramienta orientada horizontalmente en un sistema de coordenadas cilíndrico. Es de los primeros en tener un mecanismo eléctrico ya que anteriormente eran controlados por vías neumáticas.



Figura 2. 11 Robot de configuración cilíndrica modelo RT3200 por SEIKO

2.2.7 Configuración polar o esférica

Cuando el brazo de un robot es capaz de cambiar su configuración moviendo sus dos articulaciones de revolución y una articulación prismática, se denomina brazo de robot esférico o polar; la posición del brazo se describe convenientemente por medio de coordenadas esféricas; la estructura del brazo se muestra en la Figura 2.12. Los movimientos del brazo representan la rotación de la base, los ángulos de elevación y el alcance, respectivamente.

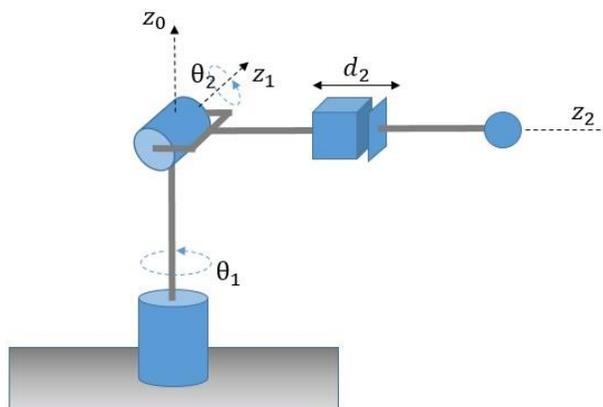


Figura 2. 12 Estructura de robot de configuración polar. [35]

Puede haber más de tres articulaciones. Sin embargo, estos tres son los básicos que forman el volumen de trabajo (Figura 2.13). Más juntas agregarían más flexibilidad, pero no cambiarían radicalmente el área accesible. Por ejemplo, si la pinza pudiera girar, sería un robot de 4 ejes, mientras que el entorno de trabajo no cambiaría en absoluto.

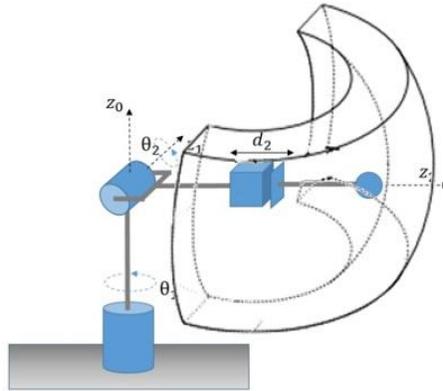


Figura 2. 13 Volumen de trabajo de robot tipo polar.

En este punto hacemos de nuevo referencia al robot Unimate manufacturado por la compañía Unimation en 1961, considerado como el primer robot industrial el cual era un robot con configuración esférica, este debutó en General Motors (Figura 2.14).

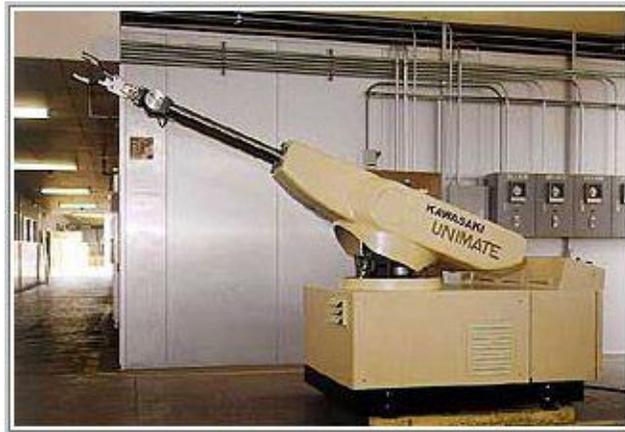


Figura 2. 14 Robot de configuración polar desarrollado por la compañía Unimation. [14]

Los brazos robóticos industriales modernos evolucionaron a partir de robots esféricos y, a veces, se los considera de este tipo, ya que su entorno de trabajo a menudo también es similar a una esfera.

2.2.8 Configuración SCARA

Es un robot con dos articulaciones de revolución y una prismática, con las dos de revolución se controla la posición respecto al plano (X, Y) y con la prismática la coordenada Z (Figura 2.15). Es rápido, barato y preciso, pero solo tiene accesibilidad a zonas de trabajo que estén en planos perpendiculares a su eje vertical (Figura 2.16). Se emplea fundamentalmente en operaciones de ensamblado o inserción de componentes electrónicos y en otros trabajos similares.

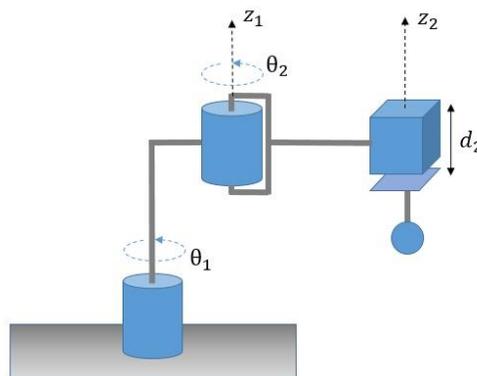


Figura 2. 15 Estructura de robot tipo SCARA. [35]

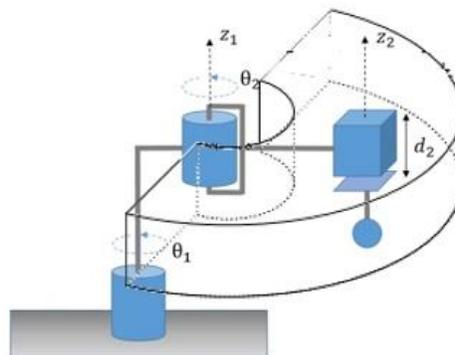


Figura 2. 16 Volumen de trabajo de robot tipo SCARA. [35]

La abreviatura SCARA significa Robot articulado para ensamblaje compatible con selectividad y es un tipo de manipulador ampliamente utilizado. Aunque el SCARA tiene una estructura RRP de las articulaciones idéntica a la del manipulador

esférico, es diferente tanto en apariencia como en el tipo de aplicaciones (Figura 2.17).



Figura 2. 17 Robot tipo SCARA desarrollado por la compañía IXA.

Desde el punto de vista de operatividad y programación, en este caso una aplicación de robot SCARA es perfectamente trasladable a un robot Cartesiano, y viceversa, ya sean procesos de manipulación, transporte, pick&place o ensamblaje entre muchos otros.

2.2.9 Configuración angular

Esta configuración está compuesta por tres articulaciones de rotación (Figura 2.18). La posición del efector final se especifica de forma natural en coordenadas angulares. El volumen de trabajo es irregular, aunque se asemeja al del robot de configuración polar (Figura 2.19).

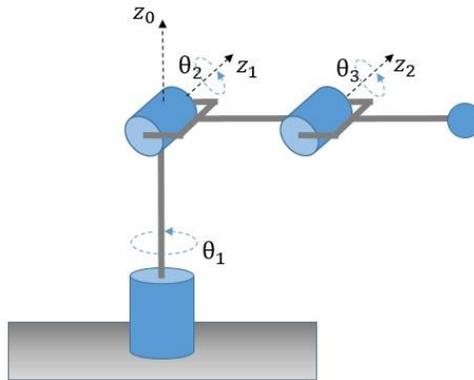


Figura 2. 18 Estructura de robot con configuración angular. [35]

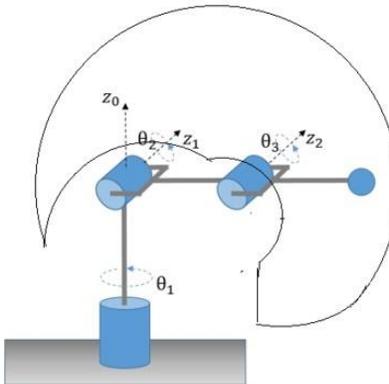


Figura 2. 19 Volumen de trabajo de robot con configuración angular. [35]

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. Este tipo de estructuras suelen emplearse mayormente en el ensamblaje automotriz.

Cabe destacar que el robot a restaurar Pegasus de AMATROL tiene este tipo de configuración, aunque tiene 5 GDL es el mismo volumen de trabajo (Figura 2.20).

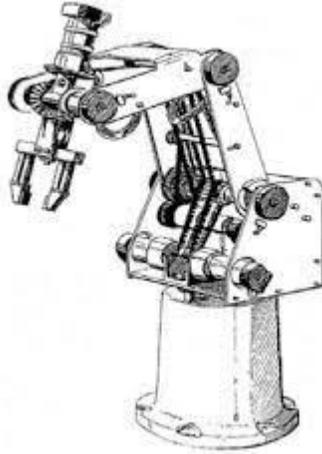


Figura 2. 20 Robot Pegasus desarrollado por AMATROL.

2.2.10 Configuración paralela

La configuración paralela consiste en cadenas cinemáticas cerradas cuyo efector final o plataforma móvil, está conectado a la base mediante varias cadenas cinemáticas seriales independientes (Figura 2.21).

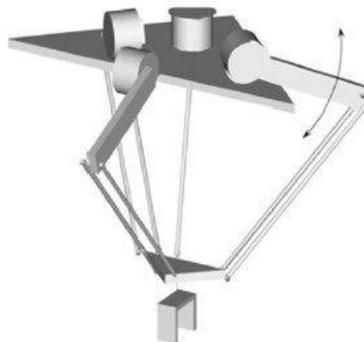


Figura 2. 21 Estructura de robot con configuración paralela. [37]

Su brazo de cuatro ejes distribuye uniformemente la carga en el robot para un transporte y montaje de alta eficiencia. La principal ventaja de los robots paralelos, respecto a los mecanismos puramente paralelos, es que el campo de movimiento rotacional es menos limitado (Figura 2.22).

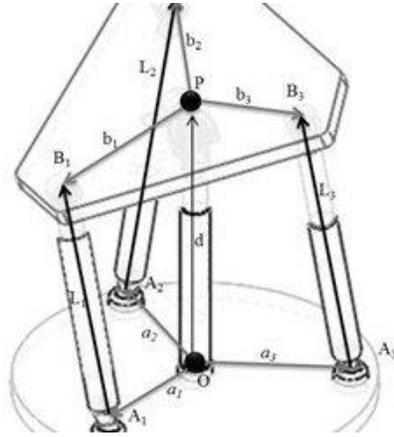


Figura 2. 22 Volumen de trabajo de robot con configuración paralela. [37]

Se usan como plataformas móviles, permitiendo simulaciones de las cabinas de vuelo de los pilotos, maquinado de piezas y transferencia de piezas a alta velocidad, podemos apreciar un robot paralelo de transferencia en la Figura 2.23.



Figura 2. 23 Robot Delta Omron Paralelo por la compañía adept.

2.3 Sensores

Los sensores permiten la medición de propiedades físicas como distancia entre objetos, velocidad de desplazamiento, fuerza etc. Estos sensores son empleados en los manipuladores para dar la realimentación que requieren los sistemas de lazo cerrado para el correcto control de las articulaciones de este. A continuación, se presentan los sensores utilizados en la restauración del robot manipulador Pegasus de AMATROL.

2.3.1 Sensor óptico reflectivo

El sensor optoreflexivo es un elemento que tiene como principal función ser un interruptor de encendido/apagado, similar a un sensor de fin de carrera.

El circuito de este sensor optoreflexivo tiene un diodo emisor infrarrojo y un fototransistor en el mismo empaque. El diodo IR es el transmisor y el fototransistor es el receptor, estos cuentan con una cubierta plástica negra con ranuras en cada extremo interno, en esta ranura pasa un haz de luz infrarroja, el empaque está diseñado para proteger el sensor de ruido de luz ambiental que pueda afectar la lectura de este. Se puede apreciar en la Figura 2.24 el diagrama de este tipo de sensor.

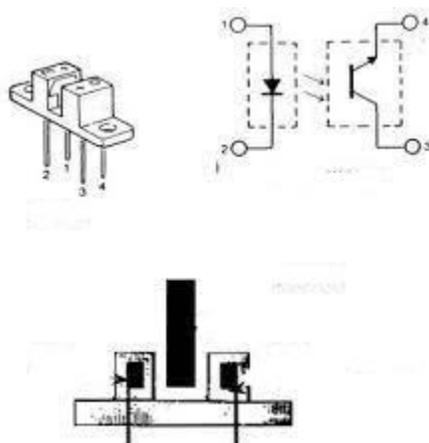


Figura 2. 24 Diagrama de un sensor optoreflexivo de herradura.

El funcionamiento de este sensor como opto interruptor se basa en las señales de salida; la salida analógica está en términos de voltaje de CC. Eso significa que cuando el fototransistor no recibe los rayos IR reflejados, la salida de voltaje es baja y cuando el fototransistor recibe rayos IR reflejados, el voltaje de salida aumenta de 0 a alrededor de 3 V, según la intensidad de la luz reflejada.

El sensor optoreflexivo también tiene salida digital. Eso significa que, cuando el fototransistor no recibe rayos IR reflejados, la salida permanece en 0 lógico (bajo), y cuando recibe luz reflejada, la salida cambia a 1 lógico (alto). Esto nos permite darle la función de sensor de seguridad ya que, dependiendo la señal, el robot continuará su rutina programada o se detendrá evitando colisiones y daños contra sí mismo, al mismo tiempo que no traspasa su volumen de trabajo.

2.4 Servomotores

Un servomotor (también llamado servo) es un dispositivo constituido por un motor de corriente continua en su núcleo, que tiene la capacidad de ubicarse en cualquier posición dada dentro de su rango de operación, y se mantiene estable en dicha posición. Al servomotor se le puede controlar tanto la velocidad como la posición, hay modelos que incluso puede controlarse el par de torque al desplazarse.

2.4.1 Constitución del servomotor

Está conformado por un motor, una caja de engranajes reductora y un circuito de control. En concreto un servomotor es un motor DC al que se ha añadido un sistema de control para manejarlo, el sistema de control está constituido por un potenciómetro, un conjunto de engranajes y una tarjeta electrónica, con lo cual conforman un sistema de lazo cerrado. Anteriormente solo se manejaban servomotores que permitían que el eje de salida solo se desplazase en un rango

de 180°; sin embargo, hoy en día los encuentras de 270° hasta 360°, los cuales son controlados su velocidad y posición en todo el rango de operación.

Los servomotores hacen uso de la modulación por ancho de pulsos (PWM) para controlar la dirección o posición de los motores de corriente continua. La mayoría trabaja en la frecuencia de los cincuenta hercios, así las señales PWM tendrán un periodo de veinte milisegundos. La electrónica dentro del servomotor responderá al ancho de la señal modulada. Si los circuitos dentro del servomotor reciben una señal de entre 0,5 a 1,4 milisegundos, éste se moverá en sentido horario; entre 1,6 a 2 milisegundos moverá el servomotor en sentido antihorario; 1,5 milisegundos representan un estado neutro para los servomotores estándares [32].

2.4.2 Tipos de controladores

El tipo de control puede ser de tres tipos: analógico, digital e híbrido en función del tipo de la señal que use el detector de error, en caso de ser un sistema en bucle cerrado, o en el regulador en caso de ser un sistema en bucle abierto. Se detalla cada uno a continuación:

- **Controles o computadores analógicos:** Las variables están representadas por ecuaciones con cantidades físicas continuas. El proceso directo de la señal analógica está ligado al uso de amplificadores operacionales y sus propiedades.
- **Controladores o computadores digitales:** Funcionan con variables discontinuas codificadas, son utilizados generalmente para la resolución de problemas referidos al funcionamiento óptimo global de una planta industrial, la toma de decisiones es una función inherente a los controladores digitales.
- **Controladores o computadores analógico-digitales:** Son los denominados controladores híbridos, los controles de funcionamiento más sofisticados suelen ser de este tipo, ya que es probable que tengan que procesar diversas señales de ambos tipos.

Las ventajas más significativas de los controladores digitales ante a los analógicos son:

- Los controladores digitales pueden realizar cálculos muy complejos a una velocidad muy alta y con el grado de exactitud que se necesite, con un coste relativamente reducido, mientras que en los analógicos el coste aumenta rápidamente ante la complejidad de los cálculos si se requiere una elevada exactitud y no es posible alcanzar la misma velocidad de resolución.
- Los controladores digitales son mucho más versátiles, simplemente cambiando el programa de aplicación, se pueden modificar absolutamente las operaciones a realizar.

2.4.3 Funcionamiento del controlador del servomotor

Cuando se envía una señal PWM al servo para que este se mueva a una posición determinada, el controlador debe calcular la distancia que debe moverse para tomar dicha posición. Este cálculo lo realiza restando la posición actual a la posición que se le ha ordenado, esta diferencia recibe el nombre de error.

La principal tarea del controlador del servo es mantener este error lo más próximo a cero que sea posible. Esto lo consigue aplicando potencia al motor interno para que el eje de salida que está acoplado al potenciómetro se mueva en la dirección necesaria para minimizar el error. Se ilustra en la Figura 2.25 el diagrama de control digital de un servomotor.

DIAGRAMA DE BLOQUE DEL SERVOMOTOR

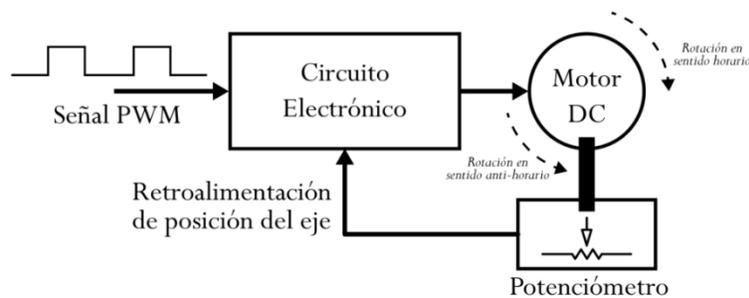


Figura 2. 25 Diagrama de bloque de servomotor digital. [39]

En un caso ideal en el que el servo dispone de un engranaje sin fricción, el par motor hará que el eje de salida se mueva ligeramente, y el valor absoluto del error aumentará a un valor diferente de cero. En estos casos el CPU integrado analiza esta condición y aplica potencia al motor para ir en contra del par motor.

En un caso real, es común que el engranaje del servo tenga cierta fricción. Esta fricción debe tomarse a consideración, dado que el par motor en proceso de medición se encuentra en realidad en el eje del motor y no en el eje de salida. El proceso de control implica conocer en todo instante la posición del rotor para lo cual se equipan los servomotores con dispositivos como encoder, potenciómetro u otros. Los mismos rotan solidariamente con el eje del servomotor e informan al servodrive la posición del rotor. Dichos dispositivos de realimentación de posición se diferencian en la robustez, resolución, capacidad de retener la información de posición ante cortes de alimentación y el número de conexiones necesarias para operar, entre otras.

Las especificaciones de los servomotores utilizados en la restauración del robot Pegasus se enlistan en las Tablas 2.1, 2.2 y 2.3.

Voltaje de operación	6V	7.4V	8.4V
Corriente al estar detenido	4mA	5mA	6mA
Velocidad de operación sin carga	0.17s/60°	0.15s/60°	0.13s/60°
Torque de parada	58Kg.cm	65Kg.cm	70kg.cm
Corriente de parada	3.5A	5A	6.2A

Tabla 2.1 Especificaciones técnicas de servomotor 60kg 5160SSG marca DS

No.	Concepto	Especificación
1	Control del sistema	PWM (Modulación por ancho de pulso)
2	Ancho del rango de pulso	500~2500µs
3	Posición neutral	1500µs
4	Grados de trabajo	270°
5	Ancho de banda muerta	3µs
6	Frecuencia de operación	50Hz
7	Dirección de rotación	Contrarreloj (500~2500µs)

Tabla 2.2 Especificaciones de operación de servomotor 60kg 5160SSG marca DS.

No.	Concepto	Especificación
1	Voltaje de operación	4.8-6.8v
2	Control del sistema	PWM (Pulso de ancho de banda)
3	Ancho del rango de pulso	500~2500µs
4	Posición neutral	1500µs
5	Grados de trabajo	270°

6	Ancho de banda muerta	3 μ s
7	Frecuencia de operación	50Hz
8	Dirección de rotación	Contrarreloj (500~2500 μ s)

Tabla 2.3 Especificaciones de operación de servomotor 35kg 8120MG marca READYTOSKY

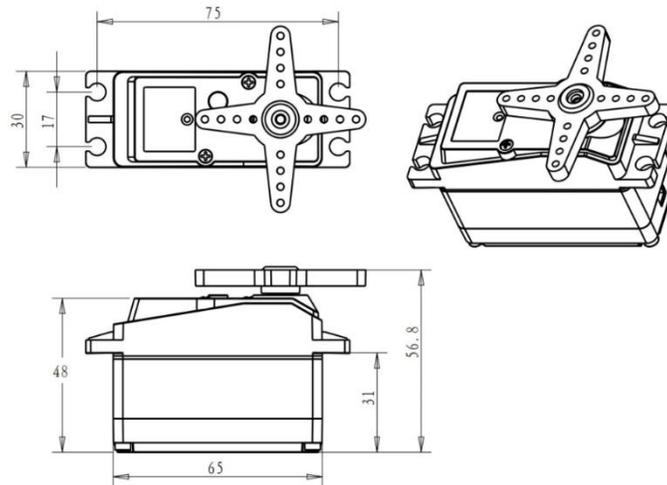


Figura 2. 26 Diagrama de dimensionamiento mecánico del servomotor 5160SSG marca DS.

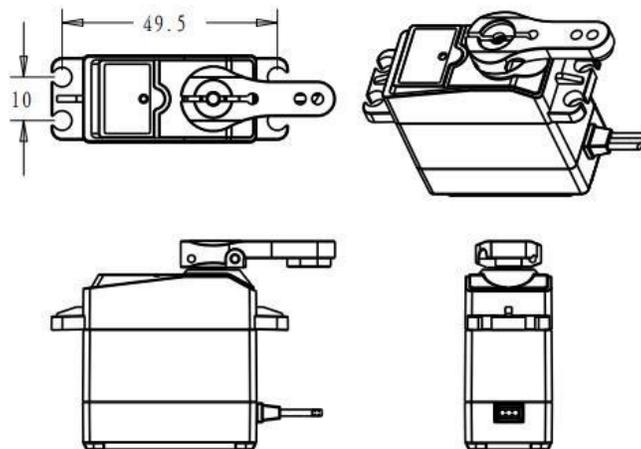


Figura 2. 27 Diagrama de dimensionamiento mecánico del servomotor 8120MG marca READYTOSKY

2.5 Sistema de Control

2.5.1 Control de posición por PWM

La señal PWM (Modulación por Ancho de Pulso) es una técnica que permite ajustar el valor medio del voltaje que va del dispositivo electrónico al actuador (generalmente motores) encendiendo y apagando la energía a alta velocidad. El voltaje promedio depende del ciclo de trabajo, o la cantidad de tiempo que la señal esté encendida en comparación con la cantidad de tiempo que la señal esté apagada en un solo período de tiempo. Para el control PWM de un servomotor el periodo permanece igual, lo que cambia es el ancho del pulso (bajo o alto) en el tiempo. Esta técnica garantiza la eficiencia al momento de reducir la velocidad ya que no reduce la potencia.

Se muestra el diagrama de relación de ciclo de trabajo, ancho del pulso y la posición de un servomotor estándar en la Figura 2.28.

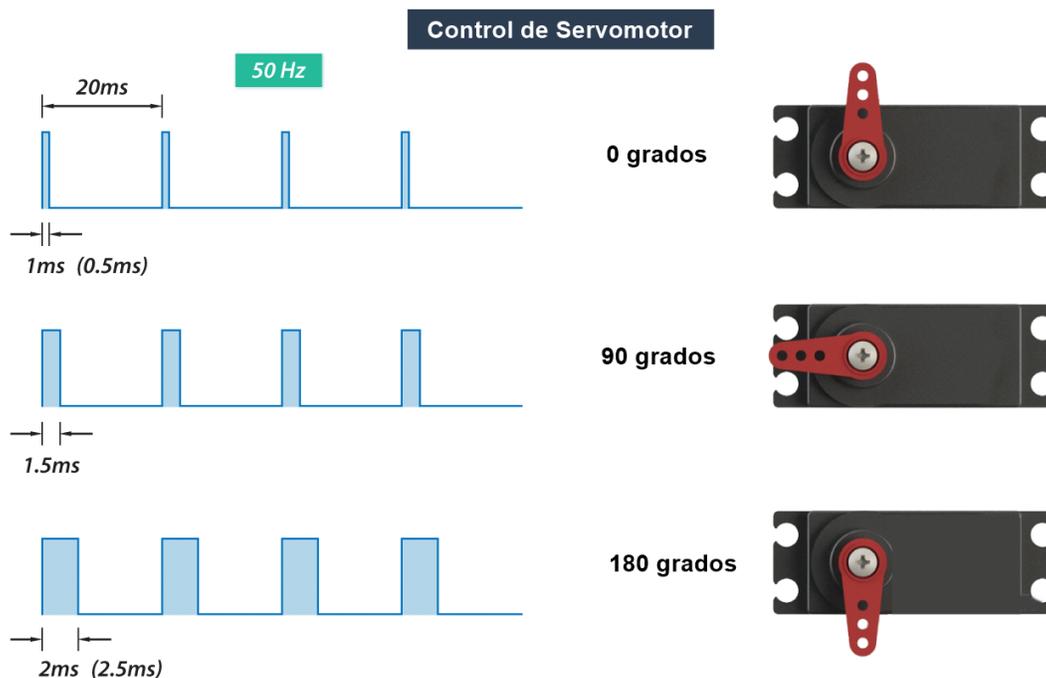


Figura 2. 28 Control de un servomotor mediante PWM. Autoría propia.

Generalmente, los pulsos con una duración de 1 ms corresponden a una posición de 0 grados, una duración de 1,5 ms a 90 grados y de 2 ms a 180 grados. Aunque la duración mínima y máxima de los pulsos a veces puede variar con diferentes marcas y pueden ser de 0,5 ms para 0 grados y 2,5 ms para la posición de 180 grados.

2.6 Consideraciones en el diseño mecánico

En este proyecto de restauración se ha realizado el remplazo de algunas piezas, para esto fue necesario el uso de la teoría del diseño, y una de las tareas más importantes del diseñador es la de imaginar la pieza o elemento que se necesita, con el fin de realizar un bosquejo de la posible pieza que satisfaga las necesidades mecánicas y de utilidad del proyecto para posteriormente hacer un análisis mecánico de esta. Una vez que se tiene claro las dimensiones y morfología de la pieza a crear se procede a tomar una de las decisiones más importantes, de la cual dependerá el buen funcionamiento de la pieza diseñada y de todo el sistema del robot, esta decisión es la de elección del material.

2.6.1 Consideraciones técnicas de diseño

El diseño y desarrollo de alguna pieza requiere no solamente el manejo de conceptos básicos de diseño mecánico convencional, sino la selección adecuada de materiales y procesos de fabricación que permitan obtener un resultado que cumpla con los requerimientos funcionales, siendo de alta calidad y con la posibilidad de manufacturarse a bajo costo. Las consideraciones técnicas de diseño que se tomaron en cuenta a la hora de fabricar una pieza para el manipulador fueron las siguientes:

- Adquisición y propiedades de los materiales (Rigidez, dureza, módulo de elasticidad)
- Aspecto del diseño

- Costos de producción
- Conocimiento del manejo de las herramientas para manufacturarlo
- Adaptabilidad

2.6.2 Materiales para engranes

Siguiendo las técnicas de diseño, el material con el que los engranes sea fabricados dependerá directamente del criterio del diseñador, considerando las características mecánicas del material (durabilidad, resistencia, desgaste por el uso), así como el método de fabricación y su asequibilidad. Con relación al proyecto, se decidió que el material a utilizar para los engranajes cónicos que se tenían que remplazar sería el PLA (ácido poliláctico) mediante impresión 3D, debido a sus propiedades mecánicas y principalmente al bajo desgaste que sufre.

El uso de engranajes de plástico se ha expandido de la transmisión de movimiento de precisión y baja potencia a aplicaciones de transmisión de potencia más exigentes. Pesan menos, menor inercia y corren mucho más silenciosamente que sus contrapartes de metal. Los engranajes de plástico a menudo no requieren lubricación o pueden combinarse con lubricantes internos como PTFE o silicona. Estos engranajes también son resistentes a muchos ambientes corrosivos. Los engranajes generalmente tienen un costo unitario más bajo que los engranajes metálicos, y pueden diseñarse para incorporar otras características necesarias en el ensamblaje [33].

Actualmente hay una gran variedad de materiales para impresión en 3D, estos vienen en la presentación de filamentos, algunos son:

- Filamento PLA: es uno de los filamentos para impresora 3D más utilizados gracias a que es muy estable y no necesita cama caliente. Se obtiene a partir de materia como el maíz y el trigo, hecho que le hace no producir gases tóxicos al contrario que otros filamentos [34].

- Filamento ABS: este tipo de filamento para impresión 3D es utilizado sobre todo en ambientes profesionales. Se caracteriza por su gran resistencia, tanto a impactos como a altas temperaturas. Además, permite realizar trabajos sobre la pieza una vez impresa. Tiene como desventaja que produce gases nocivos en el momento de la impresión, lo que dificulta la presencia de grupos de trabajo en torno a la impresora [34].
- Filamentos PETG: este filamento para impresora 3D es un plástico muy similar al PET utilizado en envases para alimentos y bebidas. Es un material muy resistente a la corrosión, la temperatura, impactos y agentes químicos. Además, es impermeable y presenta una reducida absorción de la humedad, lo que lo convierte en ideal para conservar productos de alimentación [34].
- Filamento flexible: este filamento para impresora 3D combina plástico con caucho y se caracteriza por su posibilidad de estiramiento y su resistencia a los impactos y a las roturas por maleabilidad. Además, es un filamento reciclable y su superficie presenta gran suavidad. Es conveniente no fundirlo a temperaturas superiores a las recomendadas para evitar la pérdida de elasticidad [34].
- Filamento PC: es un material termoplástico realmente resistente que soporta bien elevadas temperaturas con una alta claridad óptica. Tiende a doblarse y deformarse similar al caucho duro hasta que finalmente se rompe [34].

2.6.3 Grafica de esfuerzo-deformación del acero y aluminio

En la elaboración de un robot manipulador es importante la elección de los materiales adecuados para su estructura, es importante el uso de un material liviano ya que dependerá de ello en gran parte para su movilidad y la elección de

los motores que le darán la fuerza para moverse, por ende, involucra también los costos. En el robot Pegasus se puede apreciar que la mayor parte de la estructura está elaborada con aluminio, éste es un buen material ya que es fácil de conseguir, es relativamente económico, fácil de maquinar y más liviano que el acero. En la Figura 2.29 se muestra la gráfica de esfuerzo-deformación del aluminio, es importante tener conocimiento de estos valores característicos del material ya que solo de esta forma se podrá realizar un correcto diseño y fabricación de piezas. También se muestra la gráfica de esfuerzo-deformación del acero (Figura 2.30), esta se analizó ya que es el material utilizado para la elaboración de los acoplamientos entre el eje del servomotor y los ejes para la movilidad de los brazos del robot.

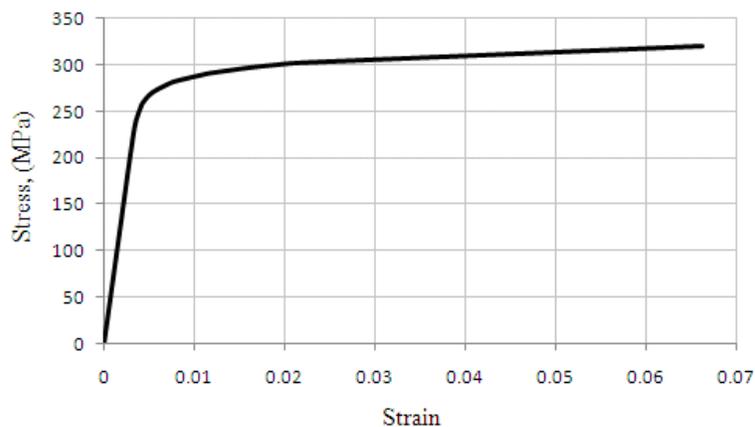


Figura 2. 29 Grafica de esfuerzo-deformación del aluminio 6020-T6 [42]

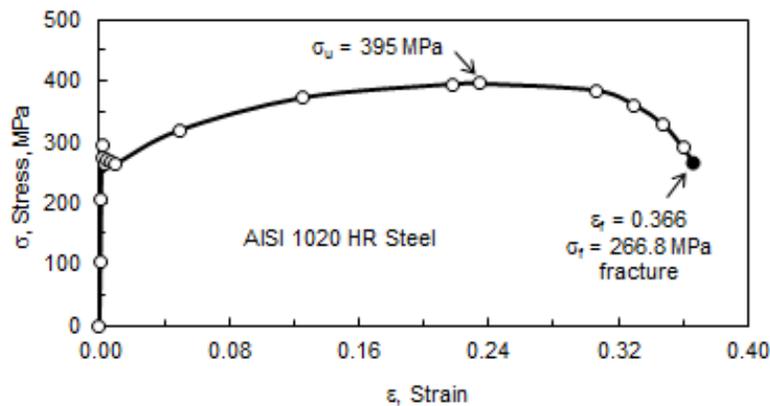


Figura 2. 30 Grafica de esfuerzo-deformación del acero AISI1020 [43]

2.6.4 Tipos de actuadores

Un actuador es un dispositivo capaz de transformar la energía hidráulica, neumática o eléctrica en energía mecánica. Los actuadores pueden considerarse transductores (elemento que convierte una forma de energía en otra); por ejemplo, el motor convierte energía eléctrica en energía mecánica.

Los elementos básicos que conforman a un actuador son:

- Sistema de accionamiento: este funciona como interruptor para iniciar o detener el movimiento.
- Sistema de transmisión: es el encargado de transmitir el movimiento del actuador a otros elementos (engranes, ejes, brazos, etc.).
- Sistema reductor: encargado de adecuar el torque y la velocidad del actuador a los valores requeridos.

Existen diferentes tipos de actuadores:

- Hidráulicos: La fuente de energía es un fluido (aceite mineral), comúnmente son cilindros hidráulicos, motores hidráulicos, válvulas hidráulicas y electrohidráulicas.
- Neumáticos: La fuente de energía es el aire a presión, los actuadores neumáticos transforman la energía acumulada en el aire comprimido en trabajo mecánico de movimiento circular o movimiento rectilíneo. Comúnmente son cilindros neumáticos, motores neumáticos, válvulas neumáticas y electroneumáticas.
- Eléctricos: La fuente de energía es la electricidad, por ejemplo, motores de corriente continua, motores de corriente alterna, motores paso a paso y servomotores.

En la Tabla 2.4 se hace una comparación de los diferentes tipos de actuadores, en el proyecto de restauración del robot Pegasus se ha elegido usar actuadores

eléctricos, específicamente servomotores. Esto debido a su precisión, potencia de torque ofrecida y facilidad de control e instalación.

Tipo de actuador	Neumático	Hidráulico	Eléctrico
Energía que utiliza	Aire a presión (5-10 bar)	Aceite mineral (50-100 bar)	Corriente eléctrica
Ejemplo	<ul style="list-style-type: none"> • Cilindros • Motor de paletas • Motor de pistón 	<ul style="list-style-type: none"> • Cilindros • Motor de paletas • Motor de pistón 	<ul style="list-style-type: none"> • Motor corriente continua • Motor corriente alterna • Motor paso a paso • Servomotor
Ventaja	<ul style="list-style-type: none"> • Bajo costo • Rapidez • Sencillos • Robustos 	<ul style="list-style-type: none"> • Rápidos • Alta capacidad de carga • Estabilidad frente a cargas estáticas 	<ul style="list-style-type: none"> • Precisos y fiables • Silenciosos • Control sencillo • Fácil instalación
Desventajas	<ul style="list-style-type: none"> • Requieren instalaciones especiales (compresor, filtro) • Ruidosos 	<ul style="list-style-type: none"> • Difícil mantenimiento • Instalación especial (filtros, 	<ul style="list-style-type: none"> • Potencia limitada

		eliminación de aire) <ul style="list-style-type: none"> • Frecuentes fugas • Costosos 	
--	--	---	--

Tabla 2.4 Comparativa de actuadores [44].

2.6.5 Transmisión de potencia

Las transmisiones son los elementos encargados de transmitir el movimiento desde los actuadores hasta las articulaciones. Se incluyen junto con las transmisiones a los reductores, estos son los encargados de adaptar el par y la velocidad de salida del actuador a los valores adecuados para el movimiento de los elementos del robot, aunque en nuestro caso específico los servomotores ya cuentan con una caja reductora para adaptar el par. En este apartado se exponen brevemente los componentes de transmisión de potencia que componen al robot Pegasus.

2.6.6 Ejes

Un eje es un elemento mecánico, generalmente cilíndrico, que sirve para sostener diferentes piezas mecánicas que giran. Realizan esfuerzos de flexión y a veces de cortadura.

Se pueden clasificar:

- Ejes fijos: permiten el giro de los elementos mecánicos situados sobre ellos, pero no giran solidariamente con ellos.
- Ejes giratorios: pueden girar solidariamente con algunos de los elementos situados sobre ellos.

Según los esfuerzos que deban soportar los ejes serán macizos o huecos. Los ejes macizos pueden llegar a deformarse si los esfuerzos de flexión son muy

grandes. Los ejes huecos soportan mejor los esfuerzos de flexión con igual cantidad de material.

En el robot Pegasus se encuentran instalados 2 ejes macizos a los cuales van fijadas diferentes piezas mecánicas como engranes y acoplamientos. También cuenta con un eje hueco en la unión de la base con fija con el hombro del brazo.

2.6.7 Engranés

Los engranajes son juegos de ruedas que disponen de unos elementos protuberantes denominados "dientes", que encajan entre sí, de manera que otras ruedas (las motrices) mediante el acoplamiento mecánico transmiten la potencia de una a otra. Una rueda dentada transmite el movimiento a la contigua que se mueve en sentido opuesto al original (Figura 2.31).

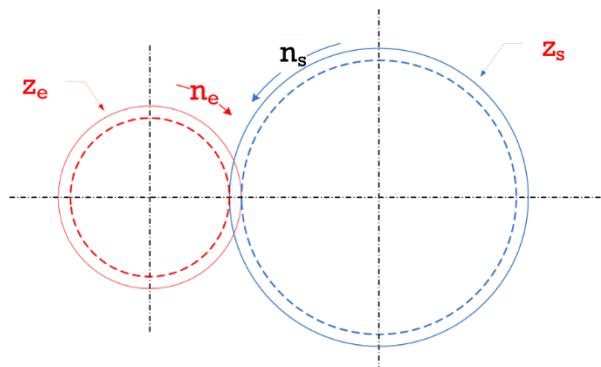


Figura 2. 31 Representación normalizada de engranajes y parámetros característicos. Autor: Manuel Torres Búa

Estos a su vez se dividen principalmente en tres tipos:

- Engranajes rectos: se utilizan en transmisiones de ejes paralelos (Figura 2.32). Son uno de los mecanismos más utilizados, y se encuentran en cualquier tipo de máquina: relojes, juguetes, herramientas, etc.

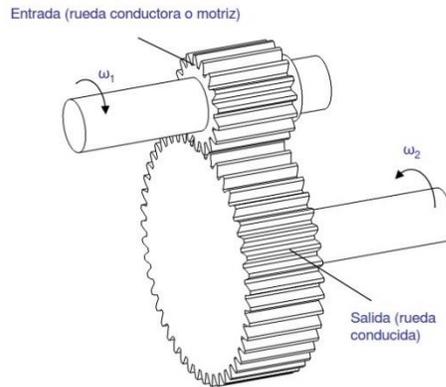


Figura 2. 32 Engranaje recto [45].

- Engranajes helicoidales: sus dientes están dispuestos siguiendo la trayectoria de hélices paralelas alrededor de un cilindro. Pueden transmitir movimiento (potencia) entre ejes paralelos o entre ejes que se cruzan en cualquier dirección (incluso perpendiculares). Este sistema de engrane proporciona una marcha más suave que la de los engranajes rectos, ya que en el mismo instante hay varios pares de dientes en contacto, lo cual hace que se trate de un sistema más silencioso, con una transmisión de fuerza y de movimiento más uniforme y segura.

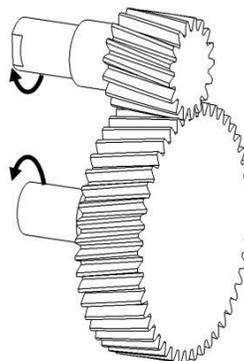


Figura 2. 33 Engranajes helicoidales [45].

- Engranajes cónicos: Se emplean para transmitir movimiento (potencia) entre ejes perpendiculares, o para ejes con ángulos distintos a 90 grados.

Se trata de ruedas dentadas en forma de tronco de cono, y pueden ser rectos o curvos (hipoides), siendo estos últimos muy utilizados en sistemas de transmisión para automóviles.

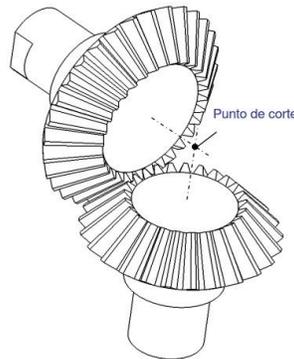


Figura 2. 34 Engranajes cónicos [45]

2.6.8 Acoplamientos

Los acoplamientos o acoples mecánicos son elementos de una máquina que sirven para prolongar líneas de transmisión de ejes o conectar secciones de diferentes ejes, en planos diferentes o con dirección paralela, para transmitir energía. Los acoplamientos pueden tener muchas funciones, pero su propósito principal es el de conectar los ejes de las unidades que fueron manufacturadas por separado y que giran, como el motor o el actuador que transmite la potencia.

Estos, sin embargo, permiten un cierto movimiento final o desalineación para la flexibilidad y también proporcionan una fácil desconexión de los dos dispositivos independientes para las reparaciones o modificaciones. Además, reducen el choque que se transmite de un eje a otro, protegen contra las sobrecargas y pueden alterar la cantidad de vibraciones que experimenta una unidad giratoria.

Están clasificados en tres tipos diferentes como lo son:

- Rígidos, que se subdividen en: rígidos de manguitos, rígidos de platillos, rígidos por sujeción cónica y rígidos de sujeción cúbica.

- Flexibles, que se clasifican a su vez en; flexibles de manguitos de goma, flexibles de disco flexible, flexibles de fuelle helicoidal, flexibles de quijada de goma, flexibles direccionales tipo Falk, flexibles de cadenas, flexibles de engrane y flexibles de muelle metálico.
- Especiales o articulados; junta eslabonada de desplazamiento lateral y junta universal, mejor conocida como cruceta o cardán.

En este proyecto se usa un acoplamiento rígido maquinados en aluminio con sujeción por medio de prisioneros para acoplar el eje de los servomotores con los ejes de los brazos (Figura 2.35).



Figura 2. 35 Acoplamientos rígidos de brida 8mm. Autoría propia.

Es altamente recomendable aplicar un adhesivo al tornillo prisionero para reducir la posibilidad de que se afloje; en la Figura 2.36 se muestra el tipo de adhesivo utilizado en este proyecto para asegurar la tornillería.



Figura 2. 36 Fijador de tornillos y tuercas utilizado en el proyecto. Autoría propia.

2.7 Subsistemas

En esta sección se analizará el robot a restaurar Pegasus para facilitar el análisis se han dividido por subsistemas:

- a) Subsistema de la base
- b) Subsistema del hombro
- c) Subsistema del codo
- d) Subsistema de la muñeca
- e) Subsistema del gripper
- f) Subsistema del controlador

2.7.1 Subsistema de la base

Este subsistema es el encargado del giro de la base, el movimiento mecánico está dado por un servomotor acoplado a una corona la cual está sujeta a un rodamiento axial, sobre este rodamiento recae todo el peso de la estructura por lo cual el servomotor no realiza demasiado esfuerzo para trasladar a la posición adecuada la estructura, cabe mencionar que este servomotor mueve la base en un rango de 270° controlado por un Arduino nano mediante PWM. La sujeción del servo se realizó mediante escuadras de acero galvanizado sujetas a la base fija del robot con unos barrenos y tornillos.



Figura 2. 37 Subsistema de la base. Autoría propia.



Figura 2. 38 Mecanismo del subsistema de la base. Autoría propia.

Se realizó la instalación de una tapa fabricada con una solera de acero, la función de esta es de proteger el cableado de control y potencia que se encuentra dentro de la base ya que anteriormente se encontraba un barreno grande donde estaba colocado un motor DC (Figura 2.39).

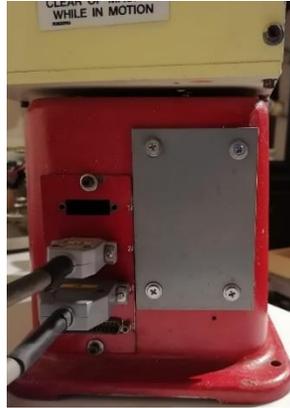


Figura 2. 39 Placa instalada en la base para el cableado. Autoría propia.

2.7.2 Subsistema del hombro

Este subsistema se encarga del movimiento del hombro, el peso de esta estructura (ejes y brazos) cae sobre 2 rodamientos ubicados a los extremos del eje de este subsistema, por ende, los ejes de los servomotores no reciben una fuerza de corte y solo necesitan aplicar el torque suficiente para mover el codo, gripper y la masa del objeto a manipular. En este subsistema se instalaron dos servomotores ya que la distancia entre el punto de apoyo y el peso aplicado (gripper y objeto a manipular) es grande, esto significa que se debe aplicar un mayor par de torque. Como se mencionó anteriormente los servomotores están acoplados al eje principal del subsistema mediante un acoplamiento rígido de platillos (Figura 2.41).



Figura 2. 40 Subsistema del hombro. Autoría propia

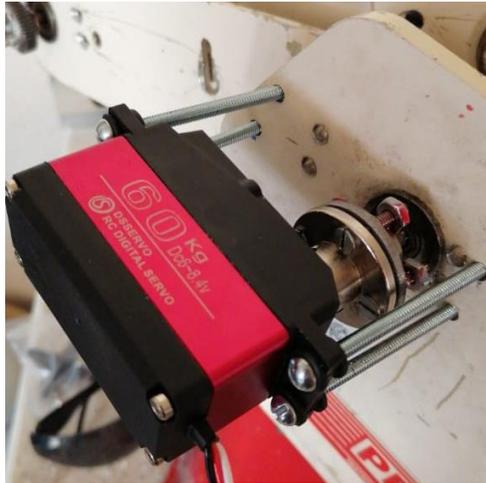


Figura 2. 41 Ilustración de acoplamiento rígido instalado en el robot. Autoría propia

2.7.3 Subsistema del codo

Este subsistema solo se encarga del movimiento del codo, la potencia se transmite mediante un servomotor acoplado al eje principal del subsistema en cada extremo, al igual que en el hombro el peso de la estructura recae en dos rodamientos, uno a cada extremo del eje. En este caso como el único peso que se necesita mover es el del subsistema del codo que está acoplado al del gripper, se instaló dos servomotores, uno de 60kg y otro 35kg.

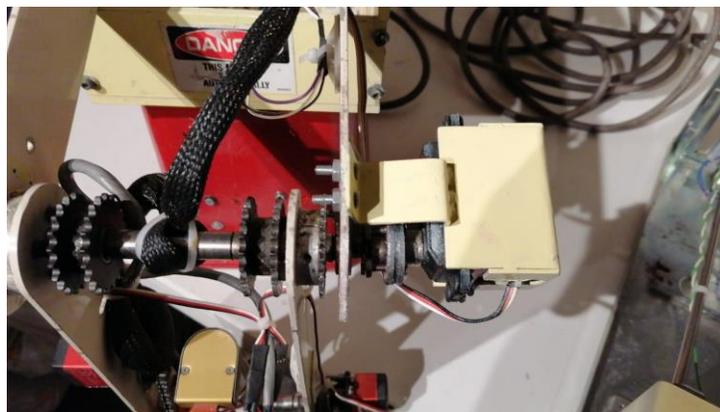


Figura 2. 42 Subsistema del codo. Autoría propia.

2.7.4 Subsistema de la muñeca

Este subsistema es un poco más complejo que los anteriores, está conformado por dos engranes cónicos (Figura 2.43), en el engrane central está montada la base del gripper, los engranes laterales se encargan de transmitir el movimiento al engrane central, cada uno de estos a su vez están acoplados a una catarina por la cual se transmite la potencia de un servomotor (a una sola catarina), con esto se logra el movimiento rotativo y hay un servomotor acoplado al eje principal de la muñeca con el cual se logra el movimiento angular. Ambos servomotores son controlados por el Arduino nano mediante PWM.



Figura 2. 43 Subsistema de la muñeca. Autoría propia.

2.7.5 Subsistema del gripper

Como se ha mencionado en anteriores capítulos, un gripper es el efector final, puede ser considerado una pinza mecánica que utiliza dedos para manipular objetos. Los dedos son la pieza más importante de los gripper ya que dependiendo el diseño podrá sujetar ciertas piezas. En el caso particular del robot Pegasus este efector final está constituido por dos dedos diseñados para manipular objetos con caras planas, objetos rectangulares, cuadrados y objetos

semiesféricos, aunque estos últimos no con tanta firmeza, se puede apreciar los dedos del gripper en la Figura 2.44.

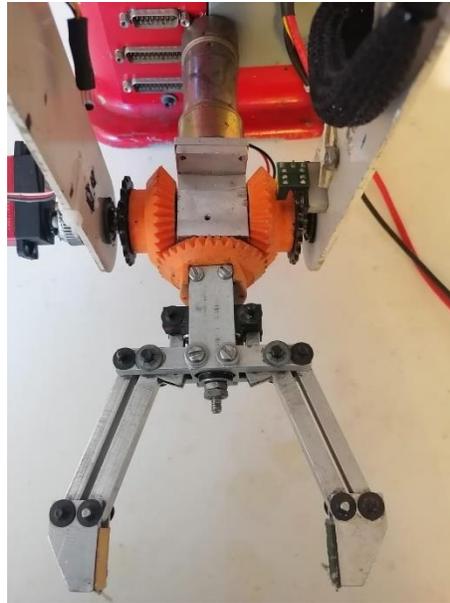


Figura 2. 44 Subsistema del gripper. Autoría propia.

El mecanismo que impulsa al gripper está constituido por un tornillo sin fin el cual enrosca en un mecanismo de tuercas, conforme el tornillo gira, los dedos del gripper abren o cierran dependiendo del giro de este. El tornillo gira ya que está acoplado a un motorreductor el cual es controlado mediante un controlador puente H, este sistema de control no da ningún tipo de realimentación al sistema de control, por ende, mediante software se trata de controlar a este como un servomotor.

2.7.6 Subsistema del controlador

Las principales operaciones de control se llevan a cabo mediante ROS el cual está instalado en una computadora, este sistema tiene comunicación con la placa electrónica que está instalada en la estructura del robot, la cual está constituida por un Arduino nano, un regulador de voltaje y un controlador puente H. La comunicación es mediante el puerto serial del Arduino y de la computadora.

CAPITULO III. DISEÑO

3.1 Introducción

En esta sección de la tesis se analizan a detalle las especificaciones de diseño: diseño de la electrónica contenida en el robot, así como diseño del control por software. En cuanto a la electrónica se refiere a los componentes que sirven para transmitir la energía eléctrica a los diferentes componentes como lo son: servomotores, el controlador puente H y el Arduino nano. El control por software es algo más complejo que el diseño de la electrónica ya que requirió de modelado en 3D, programación en lenguajes como C++ y Python, así como el manejo del software ROS para llevar a cabo el control y manejo de las articulaciones del robot mediante la comunicación serial entre ROS y la placa Arduino.

3.2 Interfaz ROS-Arduino

La interfaz que ROS ofrece al usuario es una interfaz gráfica, este es el medio por el cual el usuario tiene comunicación y control sobre el robot. Esta interfaz tiene comunicación con el robot mediante el puerto serial de la placa controladora Arduino que está conectado a través de una conexión USB2.0 a la computadora en la que está ejecutándose ROS.

3.2.1 Instalación y configuración de interfaz ROS

En esta sección se detalla como instalar una distribución de ROS en una computadora con sistema operativo Ubuntu 16.04.7, instalar un espacio de trabajo de ROS e instalar el paquete de código abierto MoveIt! para el control del robot.

Para este proyecto se ha elegido la distribución Kinetic de ROS ya que esta tiene soporte por parte de los desarrolladores en el presente año y la compatibilidad con

Linux, Mac OS y Android. Los pasos de instalación de esta distribución en Ubuntu 16.04.7 son las siguientes:

Primero se debe configurar Ubuntu para que acepte software de packages.ros.org introduciendo el siguiente comando en el terminal:

```
sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu
$(lsb_release -sc) main" > /etc/apt/sources.list.d/ros-
latest.list'
```

La siguiente línea de código es para instalar la herramienta "curl" la cual permite instalar paquetes mediante URL:

```
sudo apt install curl
```

La siguiente clave debe agregarse a Ubuntu antes de comenzar la instalación, lo que garantizará que la descarga provenga de un servidor autorizado:

```
curl -s
https://raw.githubusercontent.com/ros/rosdistro/master/ros.as
c | sudo apt-key add -
```

Esta línea de código es para que el sistema actualice las dependencias:

```
sudo apt update
```

Para ejecutar la instalación completa de la versión de ROS para escritorio se debe introducir el siguiente comando:

```
sudo apt-get install ros-kinetic-desktop-full
```

Es conveniente activar que las variables de entorno de ROS se agreguen automáticamente a la sesión de ROS cada vez se abra una nueva terminal:

```
echo "source /opt/ros/kinetic/setup.bash" >> ~/.bashrc  
source ~/.bashrc
```

RosDep es una herramienta que te permite instalar fácilmente dependencias del sistema. Para instalar esta herramienta es necesario ingresar la siguiente línea de código:

```
sudo apt install python-rosdep python-rosinstall python-  
rosinstall-generator python-wstool build-essential
```

Para inicializar RosDep se usan los siguientes comandos en la terminal:

```
sudo apt install python-rosdep  
sudo rosdep init
```

También es necesaria la instalación del paquete MoveIt!, este software de código abierto para ROS se está convirtiendo en un estándar en el campo de la robótica móvil, incluye diversas utilidades que aceleran el trabajo con brazos robóticos y ayuda a no estar continuamente “reinventando la rueda”, siguiendo la filosofía de ROS de reutilización de código.

Para la instalación de MoveIt! primero se debe instalar catkin el cual es el sistema de compilación de ROS:

```
sudo apt-get install ros-kinetic-catkin python-catkin-tools
```

La forma más sencilla de instalar MoveIt es desde binarios prediseñados con la siguiente línea de código:

```
sudo apt-get install ros-kinetic-moveit
```

Una vez seguidos estos pasos ya tenemos el espacio de trabajo necesario para poder controlar el robot manipulador mediante ROS.

3.2.2 Configuración del Arduino serial

Arduino es una plataforma de desarrollo basada en una placa electrónica de hardware libre que incorpora un microcontrolador reprogramable y una serie de pines hembra. Estos permiten establecer conexiones entre el microcontrolador y los diferentes sensores y actuadores de una manera muy sencilla [35].

En este proyecto se está usando el modelo de Arduino nano ya que el microcontrolador Atmel ATmega328 que tiene integrado es suficiente para los procesos que llevará a cabo y tiene los suficientes pines I/O para hacer las conexiones con los servomotores y demás electrónica usada.

El entorno de desarrollo integrado (IDE) de Arduino es una aplicación multiplataforma que está escrita en el lenguaje de programación Java. Esta plataforma será necesaria para cargar el código que utilizaremos para la comunicación entre ROS y el Arduino nano, podemos encontrar el paquete de instalación e instrucciones para nuestro sistema operativo desde la siguiente dirección web: <https://www.arduino.cc/en/software>

A continuación, se muestra el código utilizado para el control del robot Pegasus con comentarios en cada línea sobre su funcionamiento:

```
/*
```

```
Autor: Edrai valdiviezo Coutiño
```

Universidad de Sonora

05/11/2021

Codigo para robot manipulador Pegasus con ROS

Compilado en Arduino IDE 1.8.16

```
*/
```

```
#include <Servo.h> // Se incluye la libreria para manejo de  
servomotores
```

```
#include <ros.h> // Se incluye la libreria para comunicacion  
con ROS
```

```
#include <std_msgs/UInt16.h> // Incluimos libreria para  
manejar mensajes con el estandar ROS
```

```
#include <sensor_msgs/JointState.h> //Libreria para estar a  
la escucha de JointController
```

```
ros::NodeHandle nh; //Clase de ROS para inicio y detencion de  
comunicacion
```

```
// Se definen cada una de las articulaciones del robot como  
servos
```

```
Servo base;
```

```
Servo brazoL1;
```

```
Servo brazoL2;
```

```
Servo brazoSL;
```

```
Servo brazoSR;
```

```
Servo motorxgripper;
```

```
Servo motorygripper;
```

```
// Se inicializan las variables de los angulos de las  
articulaciones
```

```
double riel_position=0;
```

```
double base_angle= 102;
```

```

double brazoL1_angle= 22;
double brazoL2_angle= 143;
double brazoSL_angle=135;
double brazoSR_angle=135;
double motorxgripper_angle= 120;
double motorygripper_angle= 120;
double gripper_angle=30;
int posicionant=0; //toma el valor de 0 si la pinza se abrió
y 1 si se cerró
int posicionantri=0;
int mantener=0;
int EN1 = A4;
int EN2 = A5;
int EN3 = A3;
int EN4 = A2;
int enableA = 11;
int enableB = 12;
// Se recupera los valores de las articulaciones publicados
en el nodo JointStates como mensajes
// los cuales estan contenidos en un array

void servo_cb(const sensor_msgs::JointState& cmd_msg){
    riel_position=cmd_msg.position[0];
    base_angle= radianagrados(cmd_msg.position[1]); //Se asigna
a cada variable el valor recuperado del array y se envian al
metodo para convertir de radianes a grados
    base_angle=(70+base_angle);
    brazoL1_angle=radianagrados(cmd_msg.position[2]);
    brazoL1_angle= brazoL1_angle+37;
    brazoL2_angle=radianagrados(cmd_msg.position[2]);
    brazoL2_angle= (118- brazoL2_angle);
    brazoSL_angle=radianagrados(cmd_msg.position[3]);

```

```

brazoSL_angle=(74+brazoSL_angle);
brazoSR_angle=(81-brazoSL_angle);
motorxgripper_angle=radianagrados(cmd_msg.position[4]);
motorxgripper_angle=(motorxgripper_angle+79);
motorygripper_angle=radianagrados(cmd_msg.position[5]);
motorygripper_angle=(motorygripper_angle+100);
gripper_angle=radianagrados(cmd_msg.position[6]);

// Se manda la posicion a los servos de las articulaciones
base.write(base_angle);
brazoL1.write(brazoL1_angle); //para la posicion inicial debe
estar en 36, se le esta sumando 20
brazoL2.write(brazoL2_angle);
brazoSL.write(brazoSL_angle);
brazoSR.write(brazoSR_angle);
motorxgripper.write(motorxgripper_angle);
motorygripper.write(motorygripper_angle);

if(gripper_angle==20&&posicionant==0){
    digitalWrite(EN1, HIGH); //Cierra el gripper
    digitalWrite(EN2, LOW);
    analogWrite(enableA, 255); // Gire al 100% de velocidad
    delay(1650);
    digitalWrite(EN1, LOW); //Apaga el motor
    posicionant=1; //Guardamos la posicion actual del
gripper
}
else if(gripper_angle==0&&posicionant==1){
    digitalWrite(EN1, LOW); //Abre el gripper
    digitalWrite(EN2, HIGH);
    analogWrite(enableA, 255);
    delay(1600);
}

```

```

        digitalWrite(EN2, LOW); //Apaga el motor
        posicionant=0;
    }

// Codigo para controlar el motor DC del riel

if(riel_position>posicionantri){
    digitalWrite(EN3, HIGH); //Movemos hacia delante
    digitalWrite(EN4, LOW);
    analogWrite(enableB, 255); // Gire al 100% de velocidad
    int        diferencia=int(riel_position-posicionantri);
//diferencia entre posicion actual y el destino
    int tiempo=int((4000*diferencia)/1.6);
    delay(tiempo);
    digitalWrite(EN3, LOW); //Apaga el motor
    posicionantri=riel_position; //Guardamos la posicion
actual del riel
}

else if(riel_position<posicionantri){
    digitalWrite(EN3, LOW); //Movemos hacia atras
    digitalWrite(EN4, HIGH);
    analogWrite(enableB, 255);
    int diferencia=int((riel_position-posicionantri)*(-1));
    int tiempo=int((4000*diferencia)/1.6);
    delay(1100);
    digitalWrite(EN4, LOW);
    posicionantri=riel_position;
}

}

```

```

// Se declara la suscripcion al nodo joint_states en ROS
ros::Subscriber<sensor_msgs::JointState> sub("joint_states",
servo_cb);

void setup() {
nh.getHardware()->setBaud(115200); //Se especifica la
velocidad de comunicacion con ROS
nh.initNode(); //Se inicia la comunicacion
nh.subscribe(sub); // Se lleva a cabo la suscripcion al nodo

// Se asignan los servomotores de las articulaciones a los
pines fisicos del Arduino nano
base.attach(7);
brazoL1.attach(4); //L
brazoL2.attach(8); //R
brazoSL.attach(6);
brazoSR.attach(5);
motorxgripper.attach(10);
motorygripper.attach(9);
pinMode(EN1, OUTPUT);
pinMode(EN2, OUTPUT);
pinMode(enableA, OUTPUT);
pinMode(EN3, OUTPUT);
pinMode(EN4, OUTPUT);
pinMode(enableB, OUTPUT);
base.write(70);
brazoL1.write(28);
brazoL2.write(127);
brazoSL.write(15); //se debe de mover en un rango de -85 a 70
brazoSR.write(180);
motorxgripper.write(80);
motorygripper.write(110);

```

```

}

void loop() {
nh.spinOnce(); //Se ejecuta la clase de comunicacion con ROS
}

// Metodo pra convertir radianes a grados
double radianagrados(float posicion_radian)
{
    posicion_radian=int((posicion_radian*180)/3.1416);
    posicion_radian=int(posicion_radian*0.6667);
    return posicion_radian;
}

```

3.4 Tarjeta madre

Esta tarjeta es la encargada de interconectar los diferentes componentes electrónicos para el control del robot Pegasus, está ensamblada en una solera de lámina, sujetos mediante tornillos espaciadores. Contiene los conectores para la fuente de alimentación eléctrica, conector del cable USB2.0 para la comunicación con la computadora y las conexiones para cada uno de los servomotores. A continuación se enlistan los componentes electrónicos que se encuentran en la tarjeta madre:

- Placa de desarrollo Arduino nano basada en el microcontrolador ATmega328P.
- Modulo para manejar motores DC controlador puente H modelo L298N.
- Regulador Buck DC-DC step down de 15A 200W

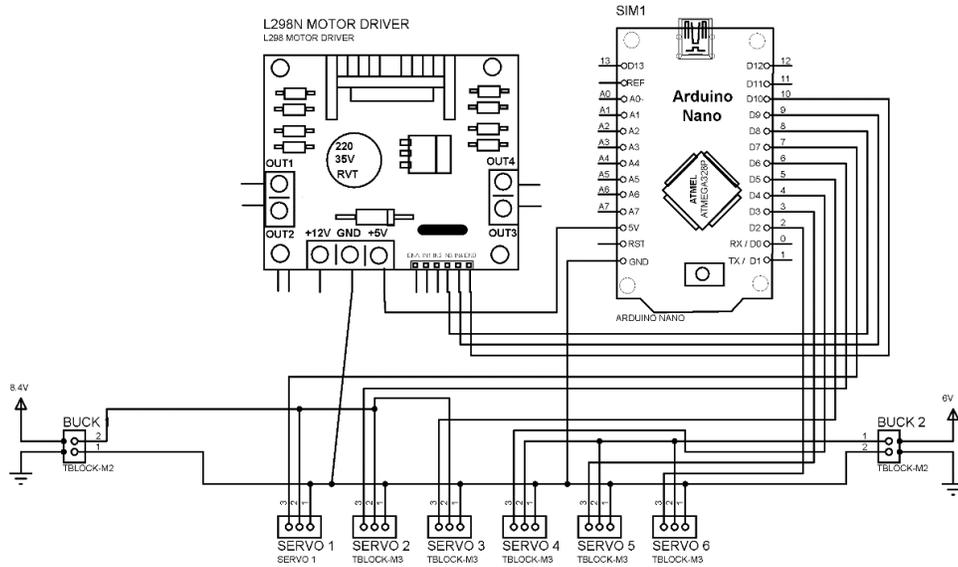


Figura 3. 1 Esquema de tarjeta madre. Autoría propia.

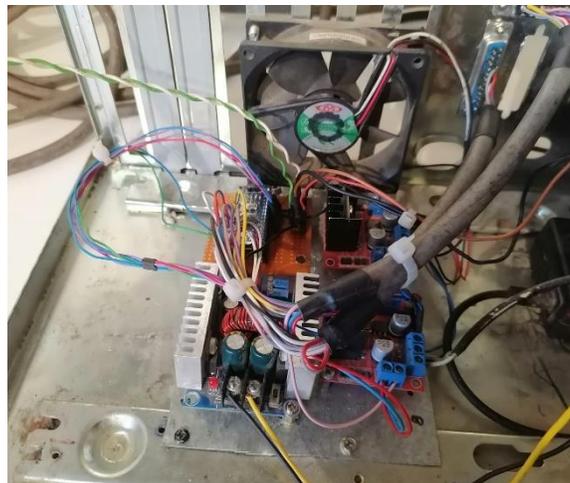


Figura 3. 2 Tarjeta madre del subsistema de control. Autoría propia.

3.5 Etapa de potencia

Para controlar este robot es necesario alimentar 6 servomotores y dos motores DC, 3 de los servomotores según las especificaciones (Tabla 2.1) tienen un torque de 60 kg/cm y tienen un voltaje de operación de 6-8.4V, otros 3 servomotores tienen un torque de 35kg/cm y un voltaje de operación de 4.5- 6V, para abrir y cerrar los dedos del gripper se utiliza un motor DC que opera en un rango de 16-

24V, al igual que el motor DC utilizado para mover la base donde va montado el robot.

Con base a las características de los motores a alimentar se seleccionaron las fuentes de poder indicadas que cumplieran con los requisitos de voltaje y amperaje que son necesarios suministrar al sistema. La fuente primaria es una fuente de poder reciclada de una computadora de escritorio la cual puede suministrar desde 3.3V hasta 16V a 25A, esta fuente está conectada a un regulador buck que controla 3 servomotores. La segunda fuente es un adaptador de corriente para laptop marca SONY que proporciona 19.5V a 4.7A.

CAPITULO IV. EVALUACIÓN ECONÓMICA

4.1 Introducción

En este capítulo se analizan y enlistan los costos generados para realizar el proyecto, es importante definir lo costos directos e indirectos ya que por medio de esto se determina la factibilidad del proyecto.

4.2 Costos indirectos

Los costos indirectos de fabricación son aquellos que si bien no se ve un impacto físico o medible en el producto final, contribuyen de una forma indirecta a la producción, estos pueden ser la energía eléctrica consumida, el costo o renta de las herramientas utilizadas para la manufactura, desgaste de equipo y administración por citar algunos ejemplos [36].

4.3 Costos directos o variables

Los costos directos son los que se relacionan estrechamente con la fabricación del producto, estos van desde las piezas, materiales que lo conforman hasta la mano de obra directa.

Las principales características de los costos directos o variables son:

- Los costos directos son gastos que pueden estar directamente relacionados con la producción de un producto y pueden incluir mano de obra directa y costos de materiales directos [37].
- Los costos directos pueden ser costos fijos como el alquiler de una planta de producción [37].

- Los costos variables varían según el nivel de producción y pueden incluir materias primas y suministros para la maquinaria [37].

4.4 Costos y presupuestos

4.4.1 Componentes electrónicos

En la siguiente tabla (Tabla 4.1) se desglosa el material que ha sido utilizado para la restauración del robot Pegasus, este comprende desde la placa fenólica hasta el cableado y consumibles como soldadura y thermofit.

Los materiales han sido cotizados y comprados en distintos establecimientos y plataformas ya que algunos componentes no son de uso común y no se encuentran en establecimientos locales. Los servomotores han sido importados de China mediante la plataforma Aliexpress y el convertidor Buck ha sido importado de Estados Unidos mediante la plataforma Ebay.

Cantidad	Descripción	Costo unitario	Total
1	Arduino Nano	\$150.00	\$150.00
1	Convertidor Buck 15A	\$300.00	\$300.00
2	Controlador motor puente H L298N	\$125.00	\$250.00
1	Placa fenólica perforada 10x10cm	\$50.00	\$50.00
1	Peineta pines macho para PCB	\$20.00	\$20.00

2	Peineta pines hembra para PCB	\$25.00	\$50.00
6	Metro de soldadura de estaño 1mm	\$10.00	\$60.00
3	Metro de cable 1x16 (AWG)	\$16.00	\$48.00
2	Metro de thermofit ¼"	\$15.00	\$30.00
1	Elementos de sujeción (Cinchos, epoxi, tornillos)	\$60.00	\$60.00
Costo total tarjeta madre			\$1,018.00

Tabla 4.1 Costos del material para elaboración de tarjeta madre. Precios (2021).

4.4.2 Componentes mecánicos

En esta sección se enlista el costo de los componentes mecánicos que han sido reemplazados o fabricados para la restauración del robot Pegasus, cabe destacar que muchos de los componentes utilizados han sido reciclados.

Cantidad	Descripción	Precio unitario	Total
1	Solera de acero de 15x8cm x ¼" con pintura industrial	\$50.00	\$50.00
12	Tornillos con tuerca de ¼" x 1"	\$2.00	\$24.00
2	Ángulos de acero galvanizado 1.5"	\$10.00	\$20.00
24	Tornillos con tuerca de ¼" por 2"	\$2.50	\$60.00

1	Juego de 4 acopladores rígidos de aluminio 8mm	\$200.00	\$200.00
1	Fijador de tuercas y tornillos marca Permatex de 10ml	\$105.00	\$105.00
1	Plastiacero marca Devcon de 25ml	\$110.00	\$110.00
3	Soportes para motores fabricados en acero inoxidable calibre 14	\$150.00	\$450.00
3	Acoplador fabricado con solera de 1/8"	\$30.00	\$90.00
3	Servomotor 35kg marca READYTOSKY	\$400.00	\$1,200.00
3	Servomotor 60kg marca DS	\$800.00	\$2,400.00
Total de costos de componentes mecánicos			\$4,709.00

Tabla 4.2 Costos de componentes mecánicos para el robot. Precios (2021).

4.4.3 Costos mano de obra e ingeniería

En esta parte se lleva a cabo el sumario de las actividades y horas hombre consumidas, así como el costo de la mano de obra, para ello ha sido necesario fijar un precio promedio a pagar por hora, este precio incluye la ingeniería y la mano de obra. El precio definido por hora en este proyecto sirve solo para dar una referencia ya que puede variar según el lugar y servicios solicitados.

No. De horas	Descripción	Precio unitario	Total
12	Diseño	\$60.00	\$720.00
2	Montaje de dispositivos	\$60.00	\$120.00
5	Proceso de soldado	\$60.00	\$300.00
1	Proceso de sujeción	\$60.00	\$60.00
Total de costos de mano de obra por tarjeta madre			\$1,200.00

Tabla 4.3 Costos de mano de obra por fabricación de tarjeta madre.

No. De horas	Descripción	Precio unitario	Total
2	Desensamble	\$60.00	\$120.00
2	Barrenado para anclaje de nuevos motores	\$60.00	\$120.00
2	Ensamble de nuevos motores en su posición	\$60.00	\$120.00
12	Pruebas y correcciones	\$60.00	\$720.00
Total de costos por mano de obra de remplazo de motores			\$1,080.00

Tabla 4.4 Costos de mano de obra por remplazo de motores.

No. De horas	Descripción	Precio unitario	Total
6	Inspección	\$60.00	\$360.00
8	Desensamble	\$60.00	\$480.00
48	Análisis mecánico	\$60.00	\$2,880.00

14	Ensamble	\$60.00	\$840.00
6	Recableado	\$60.00	\$360.00
140	Programación	\$60.00	\$8,400.00
40	Pruebas y ajustes	\$60.00	\$1,920.00
Total de costos por mano de obra de restauración del robot			\$15,240.00

Tabla 4.5 Costos de mano de obra por restauración del robot Pegasus.

Descripción	Total
Costos por mano de obra de fabricación de tarjeta madre	\$1,200.00
Costos por mano de obra de remplazo de motores	\$1,080.00
Costos por mano de obra de restauración del robot	\$15,240.00
Costo total de mano de obra para restauración del robot	\$17,520.00

Tabla 4.6 Costo total de la mano de obra del proyecto.

4.4.4 Conjunto de costos

A continuación, la sumatoria de costos de la realización del proyecto.

Descripción	Total
Total de costos de material electrónico	\$1,018.00
Total de costos de material mecánico	\$4,709.00
Total de costos por mano de obra	\$17,520.00
Costo total del proyecto	\$23,247.00

Tabla 4.7 Costo total de la realización del proyecto de restauración.

4.4.5 Comparativa con otras opciones del mercado

Para esta sección podemos tomar a consideración tres posibles opciones para llevar a cabo la tarea de la restauración del robot; la restauración se lleve a cabo por los alumnos, contratar los servicios del distribuidor el cual en este caso es AMATROL y la opción de comprar un equipo nuevo. En modo de comparativa está elaborada la Tabla 4.6 donde se muestra el costo de cada una de las opciones, el costo de rehabilitación por parte de AMATROL es en base a la cotización con el distribuidor autorizado HARRY MAZAL, S.A. DE C.V. y el costo de adquisición del equipo nuevo es en base a un promedio de precios encontrados en línea.

Descripción	Precio total
Restauración por los alumnos	\$23,247.00
Restauración por AMATROL	\$80,000.00
Adquisición de equipo nuevo	\$100,000.00

Tabla 4.8 Comparación entre opciones de restauración. Precios (2021).

CAPITULO V. RESULTADOS

Introducción

Este proyecto se ha realizado con el propósito de proporcionar un medio didáctico más a la división de ingeniería, se espera que ayude a mejorar la calidad de formación de los profesionistas egresados de la UNISON. La rehabilitación del robot Pegasus se llevó a cabo con la premisa que esta fuese de bajo costo, comparada con la compra de un equipo nuevo, como he visto en el capítulo de costos, se ahorró alrededor de un 80% el costo en comparación de la adquisición de un equipo nuevo. Como valor agregado es la incorporación del control por medio de ROS, el hardware permite hacer prácticas básicas con este software sin hacer grandes modificaciones, ideales para ejercicios en el laboratorio de robótica.

5.1 Sistema electrónico

Siguiendo con la idea de que sea un sistema fácil de reparar, que los componentes se encuentren en el mercado local y que no sean costosos, se tomó la decisión de utilizar la placa Arduino, es una de las plataformas de código abierto más utilizadas por los estudiantes, se puede adquirir con facilidad en las tiendas de electrónica, tiene una comunidad muy grande que puede brindar soporte sobre problemas en los proyectos, además que en caso de que se quisiera, puede guardar rutinas de movimiento para ser reproducido por el robot un número ilimitado de veces.

En la Figura 5.1 se muestra como fue recibido el gabinete que debería contener el sistema electrónico y eléctrico del robot Pegasus, como se puede observar la tarjeta de control electrónica está totalmente ausente, dadas las circunstancias es que se optó por hacer la placa electrónica desde cero.



Figura 5. 1 Gabinete de control original del robot Pegasus de AMATROL.

En la Figura 5.2 se aprecia el resultado de la instalación electrónica para el control del robot, la fuente de potencia original fue remplazada por una nueva debido a que no otorgaba el amperaje necesario para las operaciones del robot, además se cableó e instaló la tarjeta madre.



Figura 5. 2 Gabinete habilitado para el control del robot Pegasus de AMATROL.

5.2 Sistema eléctrico

Esta parte está constituida por las fuentes eléctricas del robot Pegasus y los motores de este, en la Figura 5.3 se puede apreciar que la fuente de potencia que se encontraba en el gabinete estaba en buen estado físico, sin embargo, esta no contaba con una fuente regulable para la alimentación de los circuitos electrónicos.



Figura 5. 3 Fuente de potencia original del robot Pegasus de AMATROL.

Los resultados de la restauración de este sistema consistieron en el remplazo de la fuente que estaba empotrada al gabinete por una fuente de poder reciclada de una computadora, la diferencia en estas es la potencia que pueden brindar al sistema, la fuente conmutada original podía brindar 5V/25A y 12V/10A, al contrario de la fuente reciclada que puede otorgar 5V/30A y 12V/26A, esto nos permitió poner en operación todos los motores del robot exitosamente.



Figura 5. 4 Fuente potencia utilizada en la restauración del robot Pegasus.

Para balancear las cargas en las fuentes se utilizó un adaptador de corriente de una laptop marca SONY como fuente de poder para alimentar el motor DC que se encuentra en el gripper, el cual tiene la función de abrir y cerrar los dedos.



Figura 5. 5 Adaptador de corriente marca SONY utilizado en la restauración del robot Pegasus.

Una pieza importante del suministro eléctrico ha sido el regulador de voltaje Buck este convierte el voltaje suministrado por la fuente de poder a uno de menor magnitud para alimentar los diferentes servomotores, en la Figura 5.6 se puede apreciar el regulador instalado gabinete del robot.

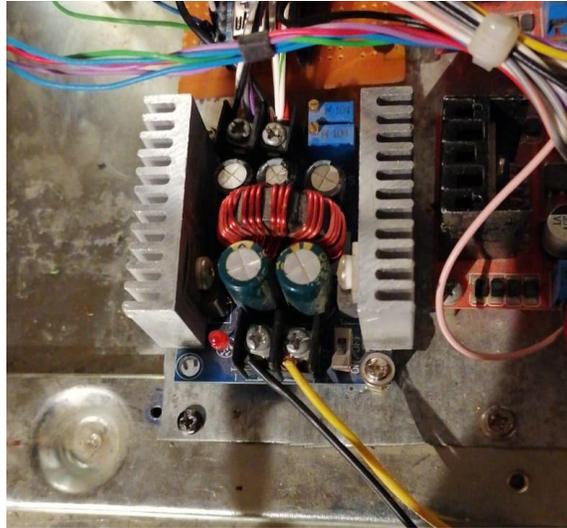


Figura 5. 6 Regulador de voltaje Buck instalado en placa madre.

5.3 Sistema mecánico

En la parte mecánica se realizaron modificaciones menores ya que se encontraban en buen estado, se retiraron las partes del sistema que ya no serían utilizadas por el cambio de motores que se realizó, se retiraron coronas, engranes, cadenas, catarinas y soportes para los motores.

Se dio mantenimiento aceitando las partes móviles como los rodamientos y dando una capa de pintura en las partes que hiciese falta para prevenir la corrosión. Se hicieron modificaciones menores en los ejes ya que se barrenaron para acoplar los ejes de los servomotores y se hicieron pequeños barrenos para unirlos con prisioneros.

Cabe mencionar que se requirió el diseño y fabricación de soportes para los servomotores ya que se requiere minimizar lo más posible las fuentes de error que perturban al sistema, al implementar estos soportes los servomotores redujeron la vibración y ganaron firmeza, en la Figura 5.7 se pueden apreciar los planos de diseño.

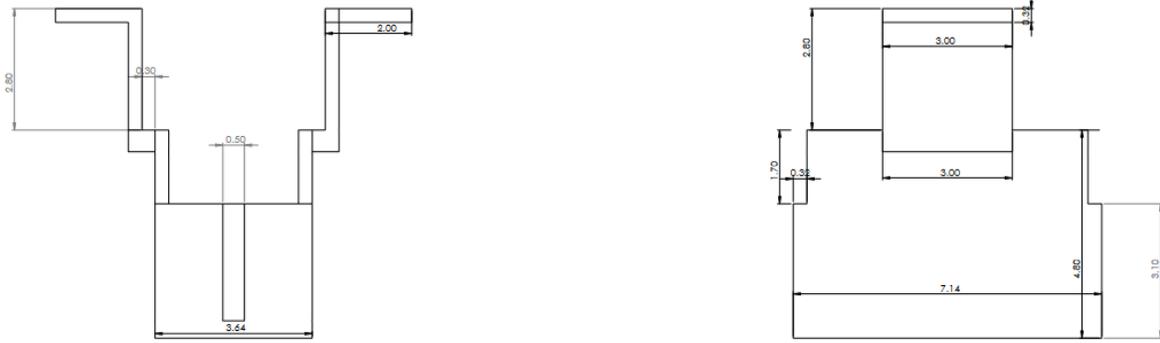


Figura 5. 7 Planos de diseño de soporte para servomotor. Autoría propia.

Una vez realizado todo lo anteriormente mencionado se procedió al reensamble del robot y el resultado final de la parte mecánica se muestra en la Figura 5.8 a) y b).

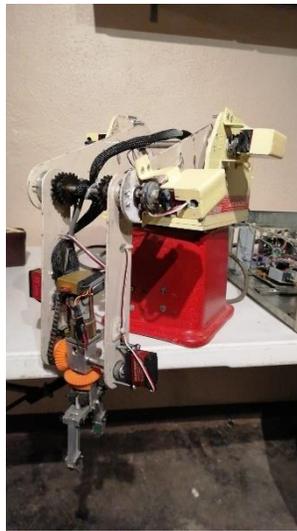


Figura 5. 8 a) Vista lateral



b) vista frontal del robot Pegasus.

5.4 Sistema de control

El sistema de control se realizó por medio de ROS, en los anexos de programación se indican los pasos para crear el archivo en formato URDF (Unified Robot Description Format) donde se especifican las dimensiones del robot, los movimientos de las articulaciones, parámetros físicos como masa e inercia, etc.

Seguidamente se explica como abrir el espacio de trabajo de Catkin y Rviz. Por medio de RViz se puede hacer lectura de sensores, procesar datos devueltos por visión estereoscópica (Cloud Point), hacer SLAM (localización y mapeo simultaneo), crear posiciones predeterminadas, crear rutinas y escenarios para simular objetos los cuales se pueden manipular con el robot o usarlos a manera de obstáculos y programar el robot para evitar colisiones.

CAPITULO VI. CONCLUSIONES

En la realización de este proyecto se presentaron grandes retos que afrontar y se obtuvo mucha experiencia de aprendizaje, la primera cuestión a analizar fue el buscar opciones de manejo de brazos manipuladores con ROS, este punto se consideró como parteaguas del proyecto ya que a partir de la información encontrada se iban a plantear las posibles soluciones para restaurar el robot y a la vez se analizaría la viabilidad de la realización, tanto en costos como el poder adquirir los componentes eléctricos y electrónicos necesarios para el desarrollo de este. Una vez se tomó la decisión de que opción de restauración era la más viable se procedió a hacer un presupuesto y a comparar este con el presupuesto de comprar un equipo nuevo o contratar una compañía autorizada que lo hiciese. Al ser mucho menor el coste de ser restaurado por un estudiante se empezó con la restauración del robot Pegasus.

Primeramente se llevó a cabo el despiece del robot para saber cómo funcionaba el mecanismo, se estudió más a fondo las piezas que lo componían y se procedió a retirar las piezas que no serían útiles en la nueva versión, se aplicaron los conocimientos adquiridos en clases como el estudio cinemático para ver el movimiento del robot en base a sus ángulos y engranes, manufactura avanzada al realizar fabricar piezas necesarias y el mantenimiento preventivo y correctivo al pintar para evitar corrosión y aceitar los rodamientos.

En la realización de la tarjeta electrónica de control se aplicaron conocimientos adquiridos en clases como de electrónica digital para la elección y programación de la placa Arduino con el microcontrolador, electrónica de potencia para la elección de las fuentes de potencia y los servomotores, robótica móvil para el análisis de los movimientos del robot y su espacio de trabajo, así como el diseño industrial para la elaboración del modelo en 3D del robot Pegasus en SolidWorks.

Como se puede observar durante el desarrollo del proyecto, las ventajas y alternativas que ofrece ROS son muy extensas a la hora de implementar este sistema a la robótica, nos permitió crear un ecosistema para el control del robot y adicionalmente una serie de posibles mejoras e integraciones de herramientas para mejorar el proyecto. El mayor reto para la elaboración del proyecto fue el aprender a manejar el sistema de ROS, ya que el conocimiento previo que se tenía sobre este era casi nulo, se optó primeramente por tomar el curso de introducción a ROS que se encuentra en la página web del sistema [29]. Seguidamente se procedió a instalar y programar los paquetes necesarios para el control del robot por medio de este software, una vez teniendo todo lo necesario en cuestiones de software y haber culminado las preparaciones de hardware en el robot se continuó con las pruebas y calibración para dar por terminada exitosamente la restauración del robot.

A manera de mencionar algunas posibles mejoras a futuro:

Gracias a que el framework MoveIt! de ROS es bastante robusto, permite pensar como futura mejora el lograr que el brazo manipulador maneje objetos por medio de visión artificial, así como el ser teleoperado por medio de conexión a internet, para estas mejoras no son necesarias grandes modificaciones más que agregar algunos complementos como una cámara estero y el desarrollo de scripts.

Referencias

- [1] L. C. y. I. C. (. O. d. E. I. p. I. Educación, Metas Educativas 2021: La educación que queremos para la generación de los bicentenarios, La Ciencia y la Cultura (OEI), 2010.
- [2] 2. e. [. 2. e. I. REAL ACADEMIA ESPAÑOLA: Diccionario de la lengua española, «REAL ACADEMIA ESPAÑOLA,» [En línea]. Available: <https://dle.rae.es/robot>. [Último acceso: 28 Septiembre 2021].
- [3] K. Capek, R.U.R (ROSSUM S UNIVERSAL ROBOTS), LONDON: PENGUIN, 1920, p. 84.
- [4] I. Asimov, I, Robot. In Runaround, New York: Spectra Books, 1942.
- [5] L. S. a. B. Siciliano, Modelado y Control de Robots Manipulador, Springer Verlag, 2000.
- [6] I. Asimov, Robots and Empire, London, England: Grafton Books, 1985.
- [7] J. J. Craig, Introducción a la mecánica y el control de la robótica, vol. 3, Addison-Wesley Publishing Company, Inc, 2006.
- [8] S. H. a. M. V. Mark W. Spong, Modelado y control de robots, New York: John Wiley & Sons, Ltd, 2006.
- [9] G. S. Bolmsjö, Industriell robotteknik, Lund,Suecia: Studentlitteratur, 1992.
- [10] J. Wallén, The History of the Industrial Robot, Linköping: Linköping University Electronic Press, 2008.
- [11] T. Koetsier, Towards the Global Intelligent Machine, Berlin/Heidelberg: Springer, 2019.
- [12] D. J. ATTORNEY, «PROGRAMMED ARTICLE TRANSFER». Estados Unidos Patente 2 988,237, 1954.
- [13] B. Malone, «George Devol: A Life Devoted to Invention, and Robots,» *IEEE Spectrum Magazine*, 2011.
- [14] J. V. Birnie, *Practical Implications of Programmable Manipulators.*, Industrial Robot: An

International Journal, 1974.

- [15] I. Zamalloa, *Dissecting Robotics—Historical Overview and Future Perspectives*, 2017.
- [16] V. D. Scheinman, *Design of a Computer Controlled Manipulator*, Stanford, CA: Stanford University, 1973.
- [17] H. & F. N. Makino, *Selective Compliance Assembly Robot Arm*, Brighton: Proceedings of 1st International Conference on Assembly Automation (ICAA), 1980.
- [18] T. K. Haruhiko Asada, «Asme Digital Collection,» July 1983. [En línea]. Available: <https://asmedigitalcollection.asme.org/vibrationacoustics/article-abstract/105/3/312/441246/Design-of-Direct-Drive-Mechanical-Arms?>. [Último acceso: 25 Septiembre 2021].
- [19] R. Clavel, *Conception d'un robot parallèle rapide à 4 degrés de liberté*, Suiza: Lausanne, EPFL, 1991.
- [20] F. R. Cortes, *Robotica Control de Robots Manipuladores*, ALFAOMEGA, 2011.
- [21] S. Bennett, *A History of Control Engineering 1930-1955*, Stevenage: Peter Peregrinus, 1993.
- [22] T. M. A. H. R. N. a. O. S. V. Falkenhahn, *Dynamic Modeling of Bellows-Actuated Continuum Robots Using the Euler–Lagrange Formalism*, IEEE, Dec. 2015.
- [23] G. B. N. a. D. T. Cielniak, *Integrating mobile robotics and vision with undergraduate computer science*, IEEE Transactions on Education, 2013.
- [24] M. & S. Shukla, *Growth of Robotics Industry Early in 21st Century*, International Journal of Computational Engineering Research (IJCER), 2012.
- [25] S. V. K. K. K. R. & D. F. Shamlan, *Fun with Robots: a Student-taught Undergraduate Robotics*, In Proceedings of the 2006 IEEE International Conference on Robotics and Automation., 2006.
- [26] B. Falah, «Pedagogical Robotics – A way to Experiment and Innovate in Educational Teaching in Morocco,» de *International Journal of Education and Learning Systems*, 2017.

- [27] S. Cousins, «Welcome to ROS Topics [ROS Topics],» IEEE Robotics Automation Magazine, 2010, pp. 13-14.
- [28] Shadow Robot Company, «Shadow Robot Company,» IEEE International Conference on Robotics and Automation, 2019. [En línea]. Available: <https://www.shadowrobot.com/>. [Último acceso: 27 09 2021].
- [29] J. E. Riva, «ROS.org,» 18 Agosto 2021. [En línea]. Available: <http://wiki.ros.org/es/ROS/Introduccion>. [Último acceso: 11 Noviembre 2021].
- [30] I. D. E. R. P. R. y. L. M. G. M. 2. r. Washington Rosell Puig, «SCIELO,» Enero 2005. [En línea]. Available: http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S0864-21412005000100008. [Último acceso: 30 Septiembre 2021].
- [31] J. P. P. ROJAS, Diseño y simulación de un raso mecánico de accionamiento hidraulico de tres grados de libertad., Escuela Politecnica Nacional de Quito, 2016.
- [32] C. A. Ramos, «CODE PROJECT,» 20 Diciembre 2012. [En línea]. Available: <https://www.codeproject.com/Articles/513169/Servomotor-Control-with-PWM-and-VHDL>. [Último acceso: 2021 Octubre 30].
- [33] J. A. V. P. J. L. Moya Rodríguez, «Cálculo de engranajes plásticos.,» de *Ingeniería Mecánica Vol. 6, No. 2*, Xalapa,Veracruz, 2007.
- [34] Mastoner, «Blog de Mastoner,» 26 Febrero 2020. [En línea]. Available: <https://www.mastoner.com/blog/post/tipos-de-filamentos-existen-para-imprimir-en-3D/>. [Último acceso: 2021 Octubre 30].
- [35] Aduino.cl, «Arduino,» 2005. [En línea]. Available: <https://arduino.cl/que-es-arduino/>. [Último acceso: 1 Noviembre 2021].
- [36] A. Pérez, «OBS Business School,» OBS Business School, 28 Abril 2021. [En línea]. Available: <https://www.obsbusiness.school/blog/costos-directos-e-indirectos-de-un-proyecto>. [Último acceso: 22 Noviembre 2021].
- [37] Corporate Finance & Accounting, «Traders studio,» 6 Abril 2021. [En línea]. Available:

- <https://traders.studio/cual-es-la-diferencia-entre-costos-directos-y-costos-variables/>. [Último acceso: 3 11 2021].
- [38] J. Pearce, «George C. Devol, Inventor of Robot Arm, Dies at 99,» *The New York Times*, p. 1, 15 Agosto 2011.
- [39] L. S. A. Gasparetto, «Semantic scholar,» 2019. [En línea]. Available: <https://www.semanticscholar.org/paper/History-of-Industrial-Robotics-in-the-20-th-Century-Gasparetto-Scalera/0a96f079d42f659effba68bb82fb882b6c66b09f/figure/1>. [Último acceso: 6 Octubre 2021].
- [40] F. Krueger, «Springer Link,» 24 Enero 2021. [En línea]. Available: <https://link.springer.com/article/10.1007/s10071-021-01472-w>. [Último acceso: 6 Octubre 2021].
- [41] C. S. Puig, «Tecnonews,» 25 Mayo 2021. [En línea]. Available: https://www.tecnonews.info/opiniones/los_robots_humanoides_algo_muy_lejano. [Último acceso: 6 Octubre 2021].
- [42] WEBMASTER, «Meccanismo complesso,» 21 Julio 2020. [En línea]. Available: <https://www.meccanismocomplesso.org/en/the-manipulators-the-most-common-configurations/>. [Último acceso: 7 Octubre 2021].
- [43] Dirección de Educación Técnica provincia de Buenos Aires, «Morfología basica de un robot industrial,» Buenos Aires, Argentina, 2020.
- [44] C. B. Khaterin Duarte, «Research gate,» Junio 2016. [En línea]. Available: https://www.researchgate.net/publication/339111741_Generalidades_de_robots_paralelos. [Último acceso: 29 Octubre 2021].
- [45] A. G. González, «Panamahitek,» 2 Diciembre 2016. [En línea]. Available: <http://panamahitek.com/que-es-y-como-funciona-un-servomotor/>. [Último acceso: 29 Octubre 2021].
- [46] K. S. V. T. P. S. Prakash Shinde, «Critical J-Integral of Thin Aluminium Sheets Employing a Modified Single Edge Plate Specimen,» Junio 2012. [En línea]. Available:

https://www.researchgate.net/publication/285691109_Critical_J-integral_of_thin_aluminium_sheets_employing_a_modified_single_edge_plate_specimen.
[Último acceso: 30 Octubre 2021].

[47] o. S. B.-G. M. A.-A. Jorge Isaías Caicedo-Reyes, *Análisis del comportamiento plástico del acero 1020 mediante ensayos de Tracción*, Riobamba, Ecuador: Escuela Superior Politécnica del Chimborazo Riobamba, Ecuador, 2020.

[48] V. R. Gonzalez, «Platea,» 2008. [En línea]. Available:
http://platea.pntic.mec.es/vgonzale/cyr_0204/cyr_01/robotica/sistema/actuadores.htm.
[Último acceso: 31 Octubre 2021].

[49] C. C. S. H. R. A. J. M. A. Juan carlos García Prada, *Teoría de máquinas*, Madrid, España, 2008.

ANEXOS

Programación

Para la programación fue necesario partir de un modelo en 3D del brazo robótico a controlar, este fue diseñado en SolidWorks y una vez listo, se exporta en formato URDF.

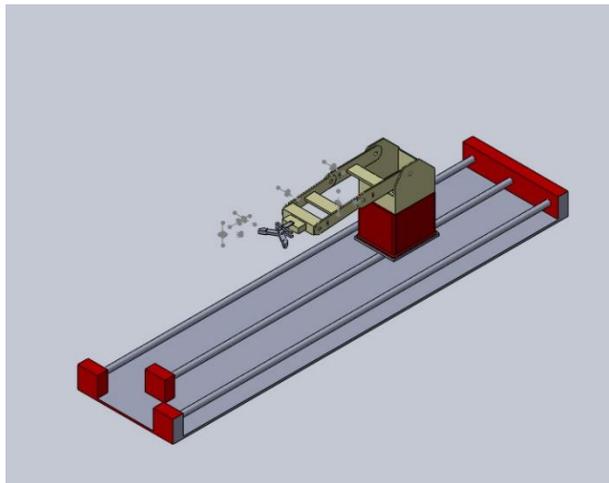


Figura A 1 Modelo en 3D del robot Pegasus realizado en SolidWorks. Autoría propia.

Al tener correctamente instalado ROS, tendremos las carpetas del espacio de trabajo:

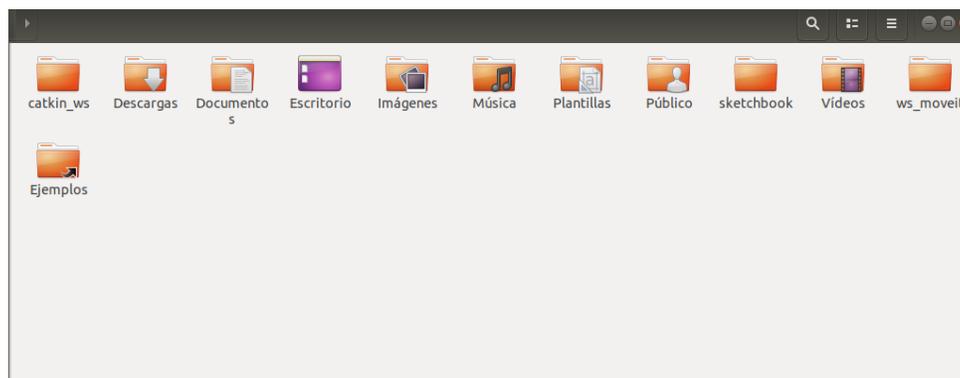


Figura A 2 Ubicación del espacio de trabajo de Catkin. Autoría propia.

A partir de haber abierto el espacio de trabajo de Catkin en una ventana de consola se puede empezar a trabajar con las herramientas de ROS.

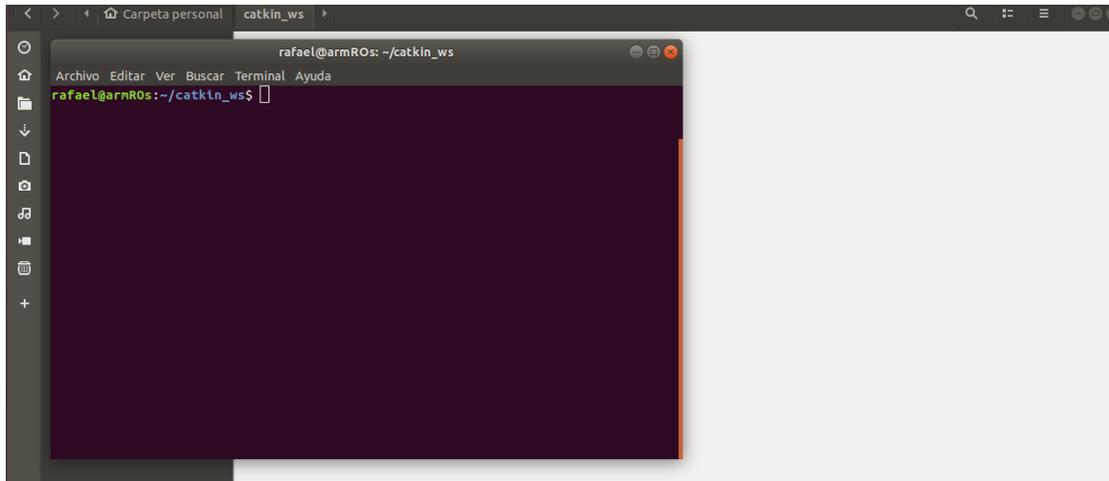


Figura A 3 Consola con el espacio de trabajo Catkin. Autoría propia.

Al ingresar los siguientes comandos podremos iniciar el asistente de MoveIt! para crear un paquete que pueda ser abierto por RViz:

```
source devel/setup.bash
```

```
roslaunch moveit_setup_assistant setup_assistant.launch
```

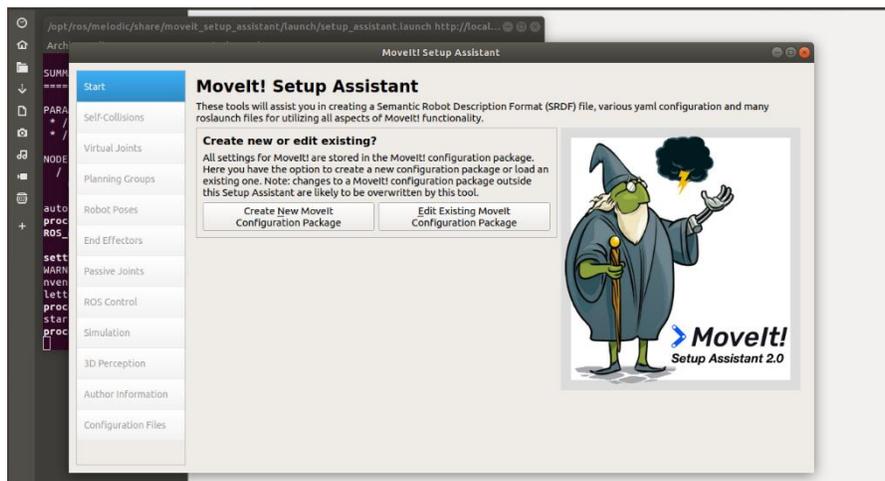


Figura A 4 El asistente de MoveIt!. Autoría propia.

Una vez cargado un archivo en formato URDF el asistente de MoveIt! te permite crear grupos para controlar con más facilidad el robot, a la derecha se previsualiza el diseño 3D del robot.

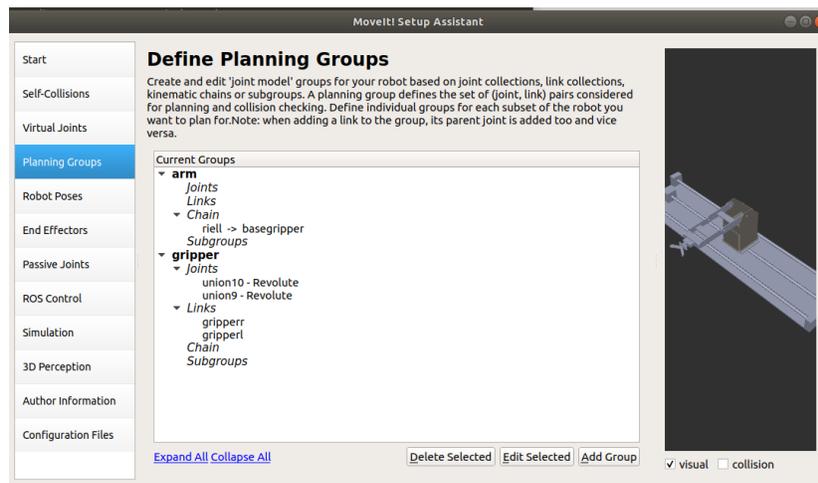


Figura A 5 División del modelo de robot en grupos. Autoría propia.

Otra de las opciones importantes a configurar es definir poses para el robot, se recomienda crear las poses de inicio y de reposo o apagado del robot.

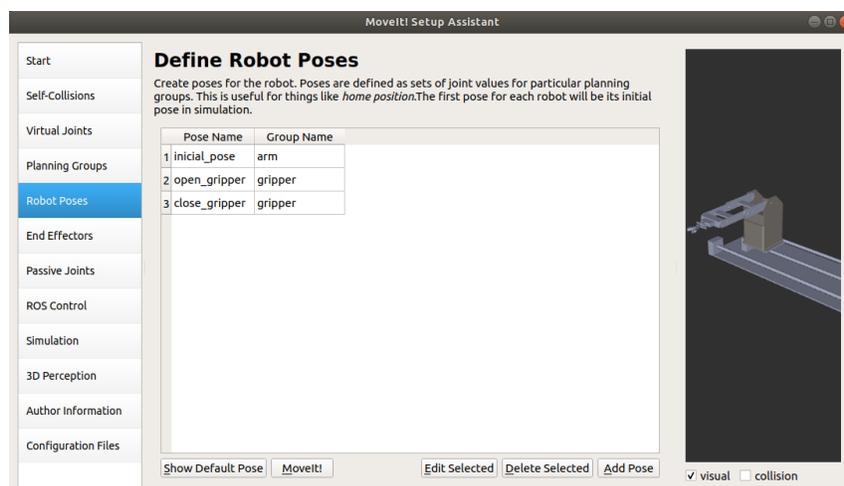


Figura A 6 Definiendo poses preestablecidas en el robot. Autoría propia.

En esta sección del asistente de MoveIt! se definen los controles que ofrece el sistema de ROS como apoyo para resolver las diferentes posiciones y movimientos que realizará el robot.



Figura A 7 Configurando los controles de ROS para el robot. Autoría propia.

Una vez que se tiene listo el paquete generado por el asistente de MoveIt! se debe abrir el paquete junto con RViz con el siguiente comando:

```
roslaunch nombre_del_paquete_generado demo.launch  
rviz_tutorial:=true
```

Aquí podemos observar el menú de RViz para crear trayectorias de movimiento.

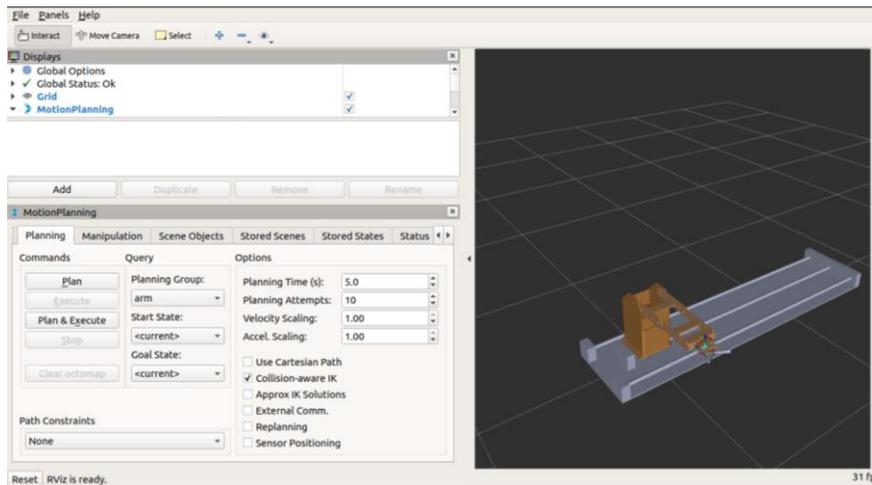


Figura A 8 Menú de planeación de movimientos MoveIt!. Autoría propia.

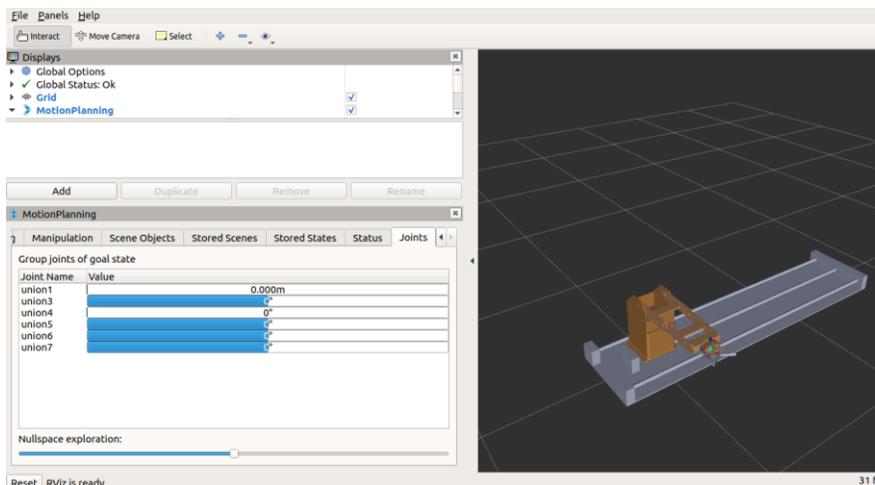


Figura A 9 Menú de control individual de los ejes del robot. Autoría propia.

Programación de script en Python para Rviz

Para que el robot ejecute una rutina automáticamente se debe programar en Python un script, el cual debe contener el nombre de los grupos de las articulaciones del robot a ser controlados y definir los puntos a los cuales se deben mover, estos deben estar definidos en coordenadas esféricas.

Para obtener las coordenadas se debe iniciar el núcleo de ROS:

```
roscore
```

Iniciamos RViz con nuestro modelo de robot:

```
roslaunch amatrol_tesis demo.launch rviz_tutorial:=true
```

Ejecutamos el comando para que ROS imprima en consola las coordenadas actuales de las uniones:

```
rostopic echo /rviz_moveit_motion_planning_display/robot_interaction_interactive_marker_topic/feedback
```

Se imprimirán en la consola las coordenadas del robot, estas se actualizan con cada movimiento del robot (Figura A10).

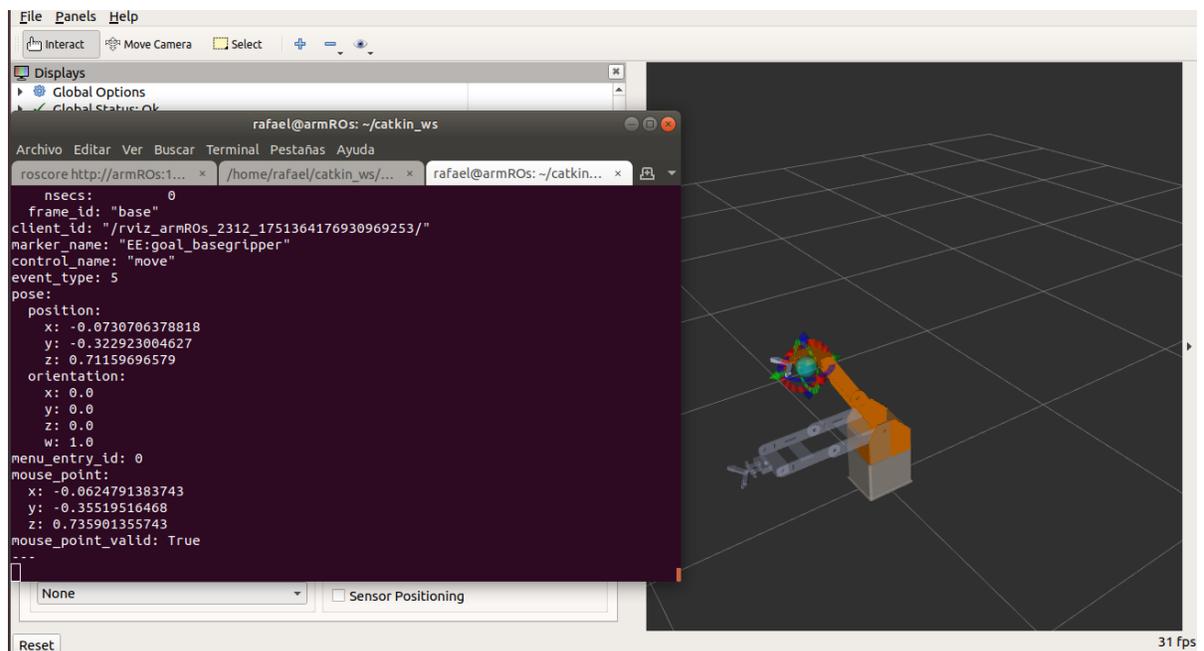


Figura A 10 Coordenadas de posición del robot.

El programa para manipular un objeto de un lugar a otro dentro del espacio de trabajo del robot sería el siguiente:

```
#!/usr/bin/env python
#importamos las librerías necesarias
import sys
import rospy
import copy, math

from moveit_commander import RobotCommander,
MoveGroupCommander
from moveit_commander import PlanningSceneInterface,
roscpp_initialize, roscpp_shutdown
from geometry_msgs.msg import PoseStamped
from moveit_msgs.msg import Grasp, GripperTranslation,
PlaceLocation, MoveItErrorCodes
from trajectory_msgs.msg import JointTrajectory,
JointTrajectoryPoint
from tf.transformations import euler_from_quaternion,
quaternion_from_euler
import random

#Debemos agregar un nombre al robot como requisito de Rviz

ROBOT_NAME = "test_robot"
if ROBOT_NAME == "test_robot":

#Definimos el nombre de los grupos
    GROUP_NAME_ARM = 'arm'
    GROUP_NAME_GRIPPER = 'gripper'

#La union entre el brazo y el gripper
```

```

GRIPPER_FRAME = 'basegripper'

#Definimos una base fija
    FIXED_FRAME = 'basefija'

#Definimos en radianes la posición abierta y cerrada del
gripper
    GRIPPER_CLOSED = 0.698
    GRIPPER_OPEN = 0.0

#Definimos el nombre de las uniones que componen el gripper
    GRIPPER_JOINT_NAMES = ['union9','union10']
    GRIPPER_EFFORT = [1.0]

#Declaramos una clase cargando la escena y grupos del robot a
RViz

class TestPick():
    def __init__(self):

        roscpp_initialize(sys.argv)
        rospy.init_node('moveit_py_demo', anonymous=True)

        scene = PlanningSceneInterface()
        robot = RobotCommander()

        right_arm = MoveGroupCommander(GROUP_NAME_ARM)
        right_gripper =
MoveGroupCommander(GROUP_NAME_GRIPPER)
        right_arm.set_goal_orientation_tolerance(0.005)
        right_arm.set_planning_time(5)
        right_arm.set_num_planning_attempts(5)

```

```

eef = right_arm.get_end_effector_link()

rospy.sleep(2)

# Activamos el replanear para que las soluciones a las
trayectorias sean mas acertadas
    right_arm.allow_replanning(True)

#Definimos el nombre de la posicion inicial

    right_arm.set_named_target("inicial_pose")
    right_arm.go(wait=True)

#Abrimos el gripper para recoger el objeto

    right_gripper.set_named_target("open_gripper")
    right_gripper.go(wait=True)

    rospy.sleep(1)

# Publicamos una escena de demostracion
    p = PoseStamped()
    p.header.frame_id = robot.get_planning_frame()

    #Agregamos un objeto para ser manipulado
    p.pose.position.x = 0.00
    p.pose.position.y = -0.38
    p.pose.position.z = 0.00
    scene.add_box("part", p, (0.07, 0.01, 0.2))

```

```

rospy.sleep(1)

start_pose = PoseStamped()
start_pose.header.frame_id = FIXED_FRAME

# Situamos al robot en la posicion de recoleccion

start_pose.pose.position.x = 0.407237410545
start_pose.pose.position.y = 0.743851780891
start_pose.pose.position.z = 0.504483044147
start_pose.pose.orientation.x = 0.0
start_pose.pose.orientation.y = 0.511886119843
start_pose.pose.orientation.z = 0.0
start_pose.pose.orientation.w = 0.859053313732
print("Moviendo a la posición de recolección")

right_arm.set_pose_target(start_pose)

#Cerramos el gripper
right_gripper.set_named_target("close_gripper")
right_arm.go(wait=True)
right_gripper.go(wait=True)

rospy.sleep(1)

right_arm.set_named_target("inicial_pose")
right_arm.go(wait=True)

#Definimos las coordenadas de la posicion de destino

next_pose = PoseStamped()
next_pose.header.frame_id = FIXED_FRAME

```

```

next_pose.pose.position.x = 0.286540865898
next_pose.pose.position.y = -0.0693459138274
next_pose.pose.position.z = 0.205837711692
next_pose.pose.orientation.x = 0.171950474381
next_pose.pose.orientation.y = 0.639209151268
next_pose.pose.orientation.z = -0.194714128971
next_pose.pose.orientation.w = 0.72383081913
right_arm.set_pose_target(next_pose)

#Abrimos el gripper para liberar el objeto

right_gripper.set_named_target("open_gripper")
right_arm.go(wait=True)
right_gripper.go(wait=True)

rospy.sleep(3)

#Volvemos el robot a la posicion inicial

right_arm.set_named_target("inicial_pose")
right_arm.go(wait=True)

rospy.spin()

#Apagamos
roscpp_shutdown()

if __name__ == '__main__':
    TestPick()

```

Conectamos el robot físico a la computadora que está ejecutando el modelo de Rviz mediante el puerto serial USB para que los comandos sean interpretados por el Arduino.

Marcamos en que puerto USB está conectado nuestro Arduino:

```
sudo chmod -R 777 /dev/ttyUSB0
```

Invocamos la librería de roserial, definimos el puerto USB y los baudios para la comunicación:

```
roslaunch roserial_python serial_node.py _port:=/dev/ttyUSB0  
_baud:=115200
```

Con esta configuración nuestro Arduino se suscribirá al tópico "joint_states" en el cual se postean las coordenadas a las que se deben de mover las articulaciones.

Control de las articulaciones individualmente por medio de ROS

Para realizar un control de los servomotores directamente desde la consola de ROS se tienen que seguir los siguientes pasos:

Iniciamos ROS abriendo una terminal desde el workspace de nuestra distribución:

```
roscore
```

Mostramos la lista de topics que se están ejecutando para elegir el de nuestro interés:

```
rostopic list
```

Ingresamos el comando para publicar manualmente en un topico, este debe estar compuesto por el nombre del topico al que queremos enviar el mensaje, la articulación que queremos mover y la posición a la que queremos mover, la cual debe estar dada en radianes:

```
rostopic pub /*topico_de_interes*//*joint_de_interes_a_mover*/  
command std_msgs/Float64 "data: 3.14159"
```