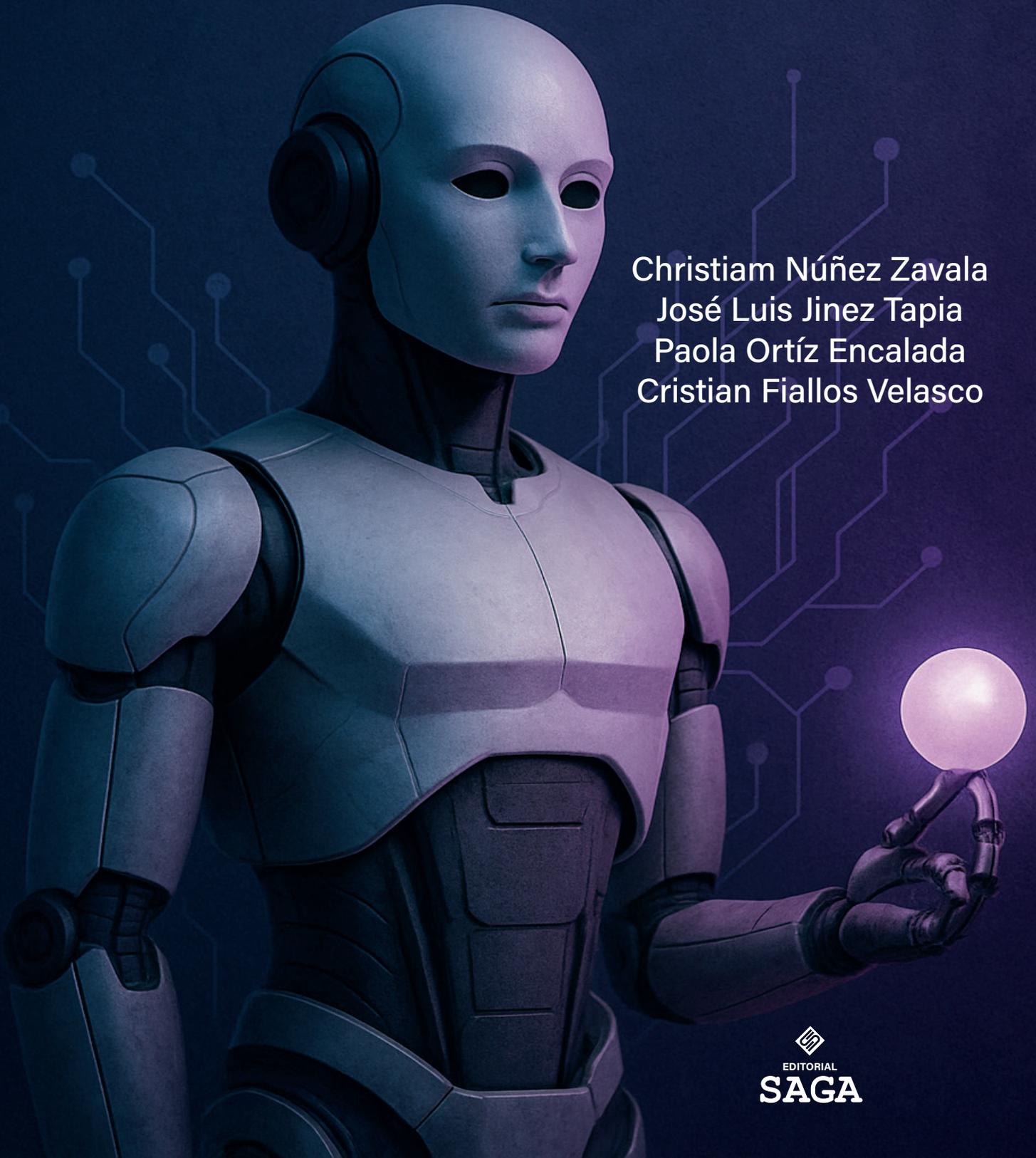


Una introducción a la robótica

Guía práctica didáctica

Christiam Núñez Zavala
José Luis Jinez Tapia
Paola Ortíz Encalada
Cristian Fiallos Velasco



Una introducción a la robótica: Guía práctica didáctica

An Introduction to Robotics: A Practical Teaching Guide

Christiam Xavier Núñez Zavala

José Luis Jinez Tapia

Paola Alexandra Ortíz Encalada

Cristian Guillermo Fiallos Velasco



**Una introducción a la robótica:
Guía práctica didáctica**

Primera edición, 2025

Núñez Zavala, Christiam Xavier

Jinez Tapia, José Luis

Ortíz Encalada, Paola Alexandra

Fiallos Velasco, Cristian Guillermo

© Derechos reservados conforme a la ley

Editorial SAGA

10 de agosto 232 entre Almendros y Mangos

Cdla. Alborada sector 2

Website: <https://libros.editorialsaga.com>

Email: editorialsaga.ec@gmail.com

Telf. (+593) 96 267 9148

Machala, Ecuador

Cubierta y diagramación: Kelvin Morales Curisaca

Dirección y supervisión editorial: Mgtr. William Satama Pereira

ISBN: 978-9942-7417-1-4

DOI: <https://doi.org/10.63415/saga.2025.26>

Impreso y hecho en Ecuador

Printed and made in Ecuador



PRESENTACIÓN-ENMARCACIÓN.....8

| | | |
|----|-------------------------------|----|
| A. | PERFIL AUTORES | 8 |
| B. | PRESENTACIÓN | 12 |
| | DESCRIPCIÓN | 12 |
| | COMPETENCIAS EDUCATIVAS | 12 |
| | OBJETIVO..... | 12 |

UNIDAD 1: ESTRUCTURA Y FUNCIONAMIENTO..... 13

| | | |
|-------|---|----|
| 1.1 | INTRODUCCIÓN A LA ROBÓTICA..... | 13 |
| 1.1.1 | ORIGEN Y DESARROLLO DE LA ROBÓTICA..... | 13 |
| 1.1.2 | DEFINICIÓN DE ROBÓT..... | 14 |

UNA INTRODUCCIÓN A LA ROBÓTICA

| | | |
|---------|--|----|
| 1.1.3 | DEFINICIÓN DE ROBÓT INDUSTRIAL MANIPULADOR..... | 15 |
| 1.1.4 | LAS TRES LEYES DE LA ROBÓTICA | 16 |
| 1.1.5 | ESTRUCTURA INTERNA DE UN ROBOT | 17 |
| 1.2 | CLASIFICACIÓN DEL ROBÓT INDUSTRIAL | 20 |
| 1.2.1 | CLASIFICACIÓN SEGÚN EL NÚMERO DE EJES (GRADOS DE LIBERTAD)..... | 20 |
| 1.2.2 | CLASIFICACIÓN SEGÚN EL TIPO DE ESTRUCTURA MECÁNICA:..... | 21 |
| 1.2.3 | CLASIFICACIÓN SEGÚN EL TIPO DE CONTROL..... | 24 |
| 1.2.4 | CLASIFICACIÓN DE ROBOTS DE SERVICIOS | 25 |
| 1.3 | ESTRUCTURA MECÁNICA DEL ROBOT. | 26 |
| 1.3.1 | GRADOS DE LIBERTAD | 27 |
| 1.3.2 | PRACTICA DE MECANISMOS CON SAM 6.0 - THE ULTIMATE MECHANISM DESIGNER | 29 |
| 1.4 | HERRAMIENTAS MATEMÁTICAS PARA LOCALIZACIÓN ESPACIAL..... | 33 |
| 1.4.1 | INTRODUCCIÓN..... | 33 |
| 1.4.2 | LOCALIZACIÓN ESPACIAL..... | 34 |
| 1.4.2.1 | <i>POSICIÓN COORDENADAS CARTESIANAS.....</i> | 35 |
| 1.4.2.2 | <i>POSICIÓN COORDENADAS POLARES Y CILÍNDRICAS.....</i> | 36 |
| 1.4.2.3 | <i>POSICIÓN COORDENADAS ESFÉRICAS.....</i> | 36 |
| 1.4.2.4 | <i>CONVERSIONES Y EJEMPLOS.....</i> | 37 |

| | |
|--|-----------|
| ACTIVIDADES UNIDAD 1..... | 41 |
| ACTIVIDAD 1: PREGUNTAS..... | 41 |
| ACTIVIDAD 2: EJERCICIOS | 42 |
| ACTIVIDAD 3: ACTIVIDADES DE SIMULACIÓN/ PRÁCTICAS..... | 43 |
| ACTIVIDAD 4: AUTOEVALUACIÓN | 43 |
| UNIDAD 2: SENSORES Y ACTUADORES..... | 46 |
| 2.1 TRANSMISORES Y REDUCTORES | 46 |
| 2.1.1 TRANSMISORES MECÁNICOS PARA ROBÓTICA..... | 46 |
| 2.1.1.1 <i>CONVERSIÓN DEL MOVIMIENTO CIRCULAR A LINEAL:</i> | 48 |
| 2.1.1.2 <i>CONVERSIÓN DEL MOVIMIENTO LINEAL A CIRCULAR:</i> | 48 |
| 2.1.2 REDUCTORES MECÁNICOS PARA ROBÓTICA..... | 49 |
| 2.1.2.1 <i>REDUCTOR HARMONIC DRIVE:</i> | 51 |
| 2.1.2.2 <i>ESTRUCTURA REDUCTOR HARMONIC DRIVE:</i> | 51 |
| 2.1.2.3 <i>CARACTERÍSTICAS REDUCTOR HARMONIC DRIVE:</i> | 52 |
| 2.2 ACTUADORES (NEUMÁTICOS, HIDRÁULICOS Y ELÉCTRICOS) | 52 |
| 2.2.1 ACTUADORES NEUMÁTICOS..... | 53 |
| 2.2.1.1 <i>CILINDROS NEUMÁTICOS</i> | 53 |
| 2.2.1.2 <i>MOTORES NEUMÁTICOS</i> | 54 |
| 2.2.2 ACTUADORES HIDRÁULICOS | 55 |
| 2.2.3 ACTUADORES ELÉCTRICOS..... | 56 |
| 2.2.3.1 <i>MOTORES DE CORRIENTE CONTINUA (DC)</i> | 57 |
| 2.2.3.2 <i>SERVOMOTOR</i> | 59 |
| 2.2.3.3 <i>MOTOR PASO A PASO</i> | 62 |
| 2.2.3.4 <i>MOTORES DE CORRIENTE ALTERNA</i> | 65 |
| 2.2.4 COMPARACIÓN ENTRE LOS DIFERENTES TIPOS DE ACTUADORES | 67 |
| 2.3 SENSORES INTERNOS (POSICIÓN, VELOCIDAD Y PRESENCIA)..... | 67 |
| 2.3.1 SENSORES DE POSICIÓN..... | 68 |
| 2.3.1.1 <i>CODIFICADORES ANGULARES DE POSICIÓN (ENCODERS)</i> | 69 |
| 2.3.1.2 <i>CAPTADORES ANGULARES DE POSICIÓN (SINCRO-RESOLVERS)</i> | 71 |
| 2.3.1.3 <i>SENSORES LINEALES DE POSICIÓN (LVDT)</i> | 73 |
| 2.3.2 SENSORES DE VELOCIDAD | 74 |
| 2.3.3 SENSORES DE PRESENCIA..... | 74 |

| | | |
|--|--|------------|
| 2.4 | ELEMENTOS TERMINALES..... | 76 |
| 2.4.1 | PINZAS (GRIPPER)..... | 78 |
| 2.4.2 | HERRAMIENTAS..... | 83 |
| <u>ACTIVIDADES UNIDAD 2.....</u> | | 85 |
| ACTIVIDAD 1: PREGUNTAS..... | | 85 |
| ACTIVIDAD 2: ACTIVIDADES DE SIMULACIÓN/ PRÁCTICAS..... | | 87 |
| <u>UNIDAD 3: PROGRAMACIÓN.....</u> | | 88 |
| 3.1 | GRADOS DE LIBERTAD GDL | 88 |
| 3.1.1 | SERVO ARDUINO: CONTROL DE POSICIÓN | 90 |
| 3.1.2 | SERVO ARDUINO: PROGRAMAR SERVOMOTOR EN ARDUINO | 92 |
| 3.1.3 | IMPLEMENTACIÓN DE CONTROL DE UN GRADO DE LIBERTAD SIMPLE (1GDL) | 94 |
| 3.1.4 | IMPLEMENTACIÓN DE CONTROL DE UN GRADO DE LIBERTAD MEDIANTE POTENCIÓMETRO (1GDL)..... | 96 |
| 3.1.5 | IMPLEMENTACIÓN DE CONTROL DE UN GRADO DE LIBERTAD MEDIANTE PULSADORES (1GDL) | 97 |
| 3.1.6 | IMPLEMENTACIÓN DE CONTROL DE DOS GRADO DE LIBERTAD MEDIANTE PULSADORES (2GDL) | 99 |
| 3.2 | CINEMÁTICA DIRECTA E INVERSA..... | 102 |
| 3.2.1 | CINEMÁTICA DIRECTA DEL BRAZO DE UN ROBOT MANIPULADOR..... | 102 |
| 3.2.2 | CINEMÁTICA INVERSA DEL BRAZO DE UN ROBOT MANIPULADOR..... | 104 |
| 3.3 | PROGRAMACIÓN DE SERVOMECANISMOS | 105 |
| 3.3.1 | PROGRAMACIÓN DE LA LECTURA ESCRITURA DE UN SERVO MECANISMO MEDIANTE UN SERVO MOTOR . | 105 |
| 3.3.2 | PROGRAMACIÓN DE LA ESCRITURA Y LECTURA DE UN SERVO MECANISMO EN LAZO CERRADO, MEDIANTE LA FUNCIÓN ESCALONAMIENTO Y LA FUNCIÓN MAP..... | 107 |
| 3.4 | PROGRAMACIÓN DE UN BRAZO MANIPULADOR | 110 |
| <u>ACTIVIDADES UNIDAD 3.....</u> | | 113 |
| ACTIVIDAD 1: PREGUNTAS..... | | 113 |
| ACTIVIDAD 2: ACTIVIDADES DE SIMULACIÓN/ PRÁCTICAS..... | | 114 |
| <u>UNIDAD 4: ROBÓTICA MÓVIL.....</u> | | 116 |
| 4.1 | LOCOMOCIÓN | 116 |
| 4.1.1 | CONSIDERACIONES DE DISEÑO..... | 117 |
| 4.1.2 | TIPOS DE LOCOMOCIÓN | 118 |
| 4.1.3 | DEFINICIONES ENTORNO A LA LOCOMOCIÓN DE UN ROBOT MÓVIL..... | 118 |

| | | |
|---------|---|-----|
| 4.1.4 | DISEÑO LOCOMOCIÓN POR RUEDAS..... | 121 |
| 4.1.4.1 | DIFERENCIAL | 121 |
| 4.1.4.2 | DISEÑO SINCRONIZADO | 122 |
| 4.1.4.3 | DISEÑO DE TRICICLO Y COCHE..... | 122 |
| 4.1.5 | DISEÑO LOCOMOCIÓN POR MEDIO DE PATAS..... | 122 |
| 4.2 | CINEMÁTICA DEL ROBOT | 123 |
| 4.2.1 | CINEMÁTICA DE ROBOTS MÓVILES | 124 |
| 4.2.2 | MODELO BÁSICO DEL MONOCICLO | 124 |
| 4.2.3 | MODELO CINEMÁTICO JACOBIANO | 125 |
| 4.2.4 | MODELOS CINEMÁTICOS DE ALGUNAS CONFIGURACIONES..... | 126 |
| 4.2.4.1 | LOCOMOCIÓN SÍNCRONA..... | 126 |
| 4.2.4.2 | LOCOMOCIÓN DIFERENCIAL | 126 |
| 4.2.4.3 | TRICICLO/BICICLETA | 127 |
| 4.3 | PERCEPCIÓN..... | 128 |
| 4.3.1 | FASE DE ADQUISICIÓN..... | 128 |
| 4.3.2 | FASE DE TRATAMIENTO..... | 128 |
| 4.3.3 | FASE DE INTERPRETACIÓN..... | 129 |
| 4.3.4 | TRANSDUCCIÓN / ENTENDIMIENTO | 129 |
| 4.4 | PLANEACIÓN DE RUTAS..... | 130 |
| 4.5 | MÉTODOS DE PLANIFICACIÓN DE TRAYECTORIAS..... | 131 |
| 4.5.1 | GRAFOS DE VISIBILIDAD:..... | 131 |
| 4.5.2 | DIAGRAMAS DE VORONOI: | 131 |
| 4.5.3 | ROADMAP PROBABILÍSTICO (PRM):..... | 133 |
| 4.5.4 | MODELADO DEL ESPACIO LIBRE: | 133 |
| 4.5.5 | DESCOMPOSICIÓN EN CELDAS: | 134 |
| 4.6 | INTRODUCCIÓN A LA VISIÓN ARTIFICIAL..... | 136 |
| 4.6.1 | PROCESOS DE VISIÓN ARTIFICIAL..... | 137 |
| 4.6.1.1 | ADQUISICIÓN..... | 137 |
| 4.6.1.2 | PROCESAMIENTO DE LA IMAGEN | 138 |
| 4.6.1.3 | REPRESENTACIÓN Y DESCRIPCIÓN (EXTRACCIÓN DE CARACTERÍSTICAS)..... | 139 |
| 4.6.1.4 | RECONOCIMIENTO E INTERPRETACIÓN | 139 |
| 4.6.2 | PRACTICAS BÁSICAS A LA INTRODUCCIÓN DE LA VISIÓN ARTIFICIAL UTILIZANDO MATLAB | 140 |
| 4.6.2.1 | LECTURA DE UNA IMAGEN DESDE EL ORDENADOR | 140 |

| | | |
|--|---|------------|
| 4.6.2.2 | LECTURA DE UNA IMAGEN DESDE EL ORDENADOR PARA SU BINARIZARLA | 141 |
| 4.6.2.3 | LECTURA DE UNA IMAGEN DESDE EL ORDENADOR Y MANEJO DE VENTANAS | 142 |
| 4.6.2.4 | PROCESO DE DESFRAGMENTACIÓN RGB DE LA IMAGEN | 143 |
| <u>ACTIVIDADES UNIDAD 4.....</u> | | 146 |
| ACTIVIDAD 1: PREGUNTAS..... | | 146 |
| ACTIVIDAD 2: ACTIVIDADES DE SIMULACIÓN/ PRÁCTICAS..... | | 147 |
| <u>BIBLIOGRAFÍA</u> | | 148 |
| BIBLIOGRAFÍA BÁSICA | | 148 |
| BIBLIOGRAFÍA COMPLEMENTARIA | | 148 |

PRESENTACIÓN-ENMARCACIÓN

A. PERFIL AUTORES

| | |
|---|--|
| Datos Autor |  |
| Nombres y apellidos | Christiam Xavier Núñez Zavala |
| Título de tercer nivel (información registrada en la SENESCYT) | Ingeniero en Electrónica, Control y Redes Industriales |
| Título de cuarto nivel (información registrada en la SENESCYT) | Magíster en Sistemas de Control y Automatización Industrial. |
| ORCID: | https://orcid.org/0000-0001-8162-5616 |
| Correos electrónico | cnunez@unach.edu.ec |
| Resumen de hoja de vida | <p>Ingeniero en Electrónica, Control y Redes Industriales de la Escuela superior Politécnica de Chimborazo en el año de 2011, estudios de posgrado Magister en Sistemas de Control y Automatización Industrial en la Superior Politécnica de Chimborazo año de 2017. Jefe de Automatización en la empresa Ecuatoriana de Cerámica planta Azulejos de la Ciudad de Riobamba. Encargado de soporte de líneas de comunicación Grupo Cerámico. Docente de la Escuela Superior Politécnica de Chimborazo en el año 2012 en la Facultad de Informática y Electrónica en las carreras de: Ingeniería Electrónica en Control y Redes Industrial, Ingeniería Electrónica en Telecomunicaciones. Docente en la Universidad Nacional de Chimborazo en la Unidad de Admisión y Nivelación en el año 2017. Analista del departamento de Tecnologías de la Información y Comunicación de la Universidad nacional de Chimborazo. Docente de la Senescyt en el Instituto Tecnológico Carlos Cisneros de la Ciudad de Riobamba en el año 2018. Actualmente Docente de la Universidad Nacional de Chimborazo en la Facultad de Ciencias de la Educación Humanas y Tecnologías, Carrera de Pedagogía de la Informática. Miembro del grupo de Investigación Umayuk de la Universidad nacional de Chimborazo. Líneas de investigación de interés: electrónica analógica, electrónica de potencia, microcontroladores, visión artificial, sistemas de control clásicos y modernos.</p> |

| | |
|---|--|
| Datos Autor |  |
| Nombres y apellidos | José Luis Jinez Tapia |
| Título de tercer nivel (información registrada en la SENESCYT) | Ingeniero en Electrónica y Telecomunicaciones |
| Título de cuarto nivel (información registrada en la SENESCYT) | Master en Ingeniería de las Telecomunicaciones, Magister en Electricidad Energías Renovables y Eficiencia Energética |
| ORCID: | 0000-0002-4113-0579 |
| Correos electrónico | jjinez@unach.edu.ec |
| Resumen hoja de vida | Master en Ingeniería de las Telecomunicaciones, Magister en Electricidad Energías Renovables y Eficiencia Energética, Ingeniero en Electrónica y Telecomunicaciones, Docente de instituciones de educación superior UNACH, Docente investigador, participación en proyectos de investigación y vinculación, publicaciones científicas director de tesis grado y posgrado, Grupo de investigación Ciber seguridad, Telemática y Tecnologías Inclusivas GI(CT)2I |

| | |
|--|--|
| Datos Autor |  |
| Nombres y apellidos | Paola Alexandra Ortíz Encalada |
| Título de tercer nivel (información registrada en la SENESCYT) | Ingeniería Industrial |
| Título de cuarto nivel (información registrada en la SENESCYT) | Maestría en Seguridad Industrial mención Prevención de riesgos y Salud Ocupacional y otra en Producción y Operaciones Industriales |
| ORCID: | https://orcid.org/0000-0001-8765-2308 |
| Correos electrónico | portiz@unach.edu.ec |
| Resumen hoja de vida | <p>Profesional en Ingeniería Industrial, con Maestría en Seguridad Industrial mención Prevención de riesgos y Salud Ocupacional y otra en Producción y Operaciones Industriales, con Diplomados en Sistemas Integrados de Gestión, Gestión del Talento Humano y Project Management, con una certificación de Auditor Interno Integral ISO 9001:2015, ISO 14001:2015 e ISO 45001:2018, con una certificación en Prevención de Riesgos Laborales y otra en Formación de Formadores, con experiencia profesional en empresas públicas y privadas de la Provincia y el País como son SECAP, Ecuatoriana de Cerámica, AZULEC, Cemento Chimborazo y experiencia en Docencia desde el año 2014 en la Carrera de Ingeniería Industrial de la Universidad Nacional de Chimborazo y en el 2022 en la UTE, de excelentes relaciones interpersonales y habilidad para trabajar en equipo, con alto grado de responsabilidad y fácil interpretación de las políticas organizacionales..</p> |

| | |
|--|--|
| Datos Autor |  |
| Nombres y apellidos | Cristian Guillermo Fiallos Velasco |
| Título de tercer nivel (información registrada en la SENESCYT) | Ingeniero en Electrónica, Control y Redes Industriales |
| Título de cuarto nivel (información registrada en la SENESCYT) | Magister en Electrónica en Automatización, Mención Redes Industriales. |
| ORCID: | https://orcid.org/0009-0005-8740-946X |
| Correos electrónico | |
| Resumen hoja de vida | <p>Magister en Electrónica en Automatización, Mención Redes Industriales de la Escuela Superior Politécnica del Ejército. Ingeniero Electrónico en Control y Redes Industriales de la Escuela Superior Politécnica de Chimborazo. Docente con diez años de experiencia en el Instituto Superior Tecnológico Carlos Cisneros, desempeñándose en la carrera de Electrónica en Automatización y Telecomunicaciones, director del grupo de investigación Fragua 2. Además, es director ejecutivo de la Academia de Robótica Educativa: Crea, Aprende y Juega, una institución dedicada a la enseñanza de nuevas tecnologías, fomentando el desarrollo de habilidades tecnológicas y creativas en niños y adolescentes.</p> |

B. PRESENTACIÓN

Descripción

Este libro trata de Conocer los fundamentos de robótica, estudiando el modelo cinemático de brazos manipuladores, los principios de la robótica móvil y sistemas de visión artificial con tecnología de hardware y software actual para diseñar, construir y programar robots prototipo descritos en la teoría.

Competencias educativas

Instalar, gestionar y proveer mantenimiento en sistemas electrónicos, control y telecomunicaciones, mediante el estudio de elementos semi-conductores, aplicada a la automatización industrial, así como para la transmisión y recepción de señales en comunicaciones analógicas y digitales.

Configurar, instalar y mantener; elementos, equipos y sistemas de control industrial con fundamentos científicos, tecnológicos y de gestión, demostrando solidez en sus valores éticos, morales y científicos.

Objetivo

Conocer los fundamentos de robótica, estudiando el modelo cinemático de brazos manipuladores, los principios de la robótica móvil y sistemas de visión artificial con tecnología de hardware y software actual para diseñar, construir y programar robots prototipo descritos en la teoría.

UNIDAD 1: ESTRUCTURA Y FUNCIONAMIENTO

1.1 Introducción a la Robótica.

1.1.1 Origen y desarrollo de la robótica.

Los primeros dispositivos que responden al que hoy se conoce como robot, no adoptaron inicialmente esta denominación, los progenitores más directos de los robots fueron los manipuladores teleoperados (Figura 1). En 1948 R. C. Goertz del Argonne National Laboratory desarrolló, con el objetivo de manejar elementos radioactivos sin riesgo para el operador, el primer sistema de telemanipulación. Éste consistía en un dispositivo mecánico maestro-esclavo (Barrientos, 2007).



Figura 1. Preparación del aire

Fuente: (Barrientos, 2007)

El manipulador maestro, situado en la zona segura, era movido directamente por el operador, mientras que el esclavo, situado en contacto con los elementos radioactivos y unido mecánicamente al maestro, reproducía fielmente los movimientos de éste. El operador además de poder observar a través de un grueso cristal el resultado de sus acciones, sentía a través del dispositivo maestro las fuerzas que el esclavo ejercía sobre el entorno (Barrientos, 2007).

Años más tarde, en 1954, Goertz hizo uso de la tecnología electrónica y del servocontrol sustituyendo la transmisión mecánica por otra eléctrica y desarrollando así el primer sistema de telemanipulación con servocontrol bilateral. Junto a la industria nuclear, a lo largo de los años sesenta la industria submarina comenzó a interesarse por el uso de los telemanipuladores (Barrientos, 2007).

A este interés se sumó la industria espacial en los años setenta. La sustitución del operador por un programa de ordenador que controlase los movimientos el manipulador dio paso al concepto de robot, espoleado por las necesidades de automatización de las cadenas de fabricación (Barrientos, 2007).

En los años finales de la década de los sesenta y principios de la década de los setenta se ponen las bases de la investigación en robótica en las universidades. Se crean departamentos de investigación en inteligencia artificial, y se diseñan los primeros robots móviles con un cierto grado de autonomía, como Shakey por el Stanford Research Institute o el Stanford Cart de la Universidad de Stanford. También se ponen las bases de los diseños actuales de brazos manipuladores, principalmente por Victor Scheinman, con el diseño del brazo de Stanford. Este robot fue el primer manipulador controlado por computador y con accionamiento eléctrico, y que llevaría más tarde al diseño, fabricación y comercialización del robot PUMA (Figura 2), uno de los robots industriales más famosos y utilizados tanto en ambientes industriales como de investigación. (Barrientos, 2007).

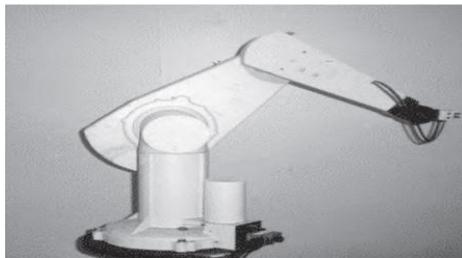


Figura 2. Preparación del aire

Fuente: (Barrientos, 2007)

1.1.2 Definición de Robót.

El gran dinamismo que los avances en robótica y disciplinas afines tienen en nuestros días origina que el concepto de robot deba ser revisado y ampliado con frecuencia. La relectura de los primeros textos literarios de ciencia ficción que tratan la robótica, presentan casas robotizadas [BRADBURY-50], automóviles robots [ASIMOV-53] o robots dotados de una alta capacidad de

interacción con los humanos [ASIMOV-54] . Todos ellos han dejado de ser hoy en día quimeras de la ciencia-ficción para ser realidades tecnológicas que entran dentro de la disciplina de la robótica. (Barrientos, 2007).

Es por ello, complicado acotar lo que debe ser entendido por un robot y consecuentemente definir este término con la suficiente generalidad como para cubrir el amplio campo de dispositivos que como tal son, hoy en día, reconocidos. En este sentido, puede resultar válido el revisar las definiciones, no estrictamente técnicas, contenidas en enciclopedias, antes de dar una definición de robot. En la Tabla 1 se recogen algunas de las definiciones consideradas por los diccionarios y enciclopedias de más prestigio: (Barrientos, 2007).

Tabla 1
Algunas definiciones de robot.

| |
|---|
| Enciclopedia Británica |
| Máquina operada automáticamente que sustituye el esfuerzo de los humanos, aunque no tiene por qué tener apariencia humana o desarrollar sus actividades a la manera de los humanos. |
| Diccionario Merrian Webster |
| Máquina que se asemeja a los humanos y desarrolla como ellos tareas complejas como andar o hablar. Un dispositivo que desarrolla de manera automática tareas complicadas, a menudo de manera repetitiva Un mecanismo guiado por control automático. |
| Diccionario de la Real Academia Española |
| Máquina o ingenio electrónico programable, capaz de manipular objetos y realizar operaciones antes reservadas sólo a las personas. |

Fuente: (Barrientos, 2007)

1.1.3 Definición de robót industrial manipulador

Los primeros intentos de establecer una definición formal de robot, surgen en el año 1979 por parte de la RIA (Robot Institute of America, actualmente Robotic Industries Association), según la cual: Robot (RIA): Un robot industrial es un manipulador multifuncional reprogramable, capaz de mover materias, piezas, herramientas o dispositivos especiales, según trayectorias variables, programadas para realizar tareas diversas (Barrientos, 2007).

Esta temprana definición, matizada y acotada, ha sido la referencia para las sucesivas definiciones que se han ido dando al robot hasta llegar a la actual, establecida por la Asociación Internacional de Estándares (ISO). Ésta en su norma ISO 8373 (en España corresponde a la UNE EN ISO

8373:1998. «Robots Manipuladores Industriales. Vocabulario»), define al Robot manipulador industrial como

Robot manipulador industrial (ISO): Manipulador de 3 o más ejes, con control automático, reprogramable, multiaplicación, móvil o no, destinado a ser utilizado en aplicaciones de automatización industrial. Incluye al manipulador (sistema mecánico y accionadores) y al sistema de control (software y hardware de control y potencia) (Barrientos, 2007).

1.1.4 Las tres leyes de la robótica

Las tres leyes de la robótica son un conjunto de normas elaboradas por el escritor de éxito Isaac Asimov que se aplican a la mayoría de los robots de sus obras y que están diseñados para cumplir órdenes. En ese universo, las leyes son «formulaciones matemáticas impresas en los senderos positrónicos del cerebro» de los robots (líneas de código del programa que regula el cumplimiento de las leyes guardado en la memoria principal de aquellos). Aparecidas por primera vez en el relato «Círculo vicioso» (Runaround, 1942), establecen lo siguiente (Asimov, 2008):

1. Un robot no hará daño a un ser humano ni, por inacción, permitirá que un ser humano sufra daño.
2. Un robot debe cumplir las órdenes dadas por los seres humanos, a excepción de aquellas que entren en conflicto con la primera ley.
3. Un robot debe proteger su propia existencia en la medida en que esta protección no entre en conflicto con la primera o con la segunda ley.¹

Esta redacción de las leyes es la forma convencional en la que los humanos de las historias las enuncian; su forma real sería la de una serie de instrucciones equivalentes y mucho más complejas en el cerebro del robot (Asimov, 2008).

Las tres leyes aparecen en un gran número de historias de Asimov, tanto en su serie de los robots como en varias historias relacionadas, y la serie de novelas protagonizadas por Lucky Starr. También han sido utilizadas por otros autores cuando han trabajado en el universo de ficción de Asimov, y son frecuentes las referencias a ellas en otras obras, no solo de ciencia ficción, sino también de otros géneros (Asimov, 2008).

1.1.5 Estructura interna de un robot

Un robot está formado por: (Salazar, s. f.).

- Sistema de control (sistema nervioso)
- Sensores
- Efectores y actuadores
- Sistema de locomoción/manipulación

1.1.5.1 Sistema de control (sistema nervioso)

La función de este sistema es controlar las acciones que ejecuta el robot de tal forma que pueda cumplir con la tarea que le ha sido asignada y tomando en consideración la información del medio ambiente (Salazar, s. f.).

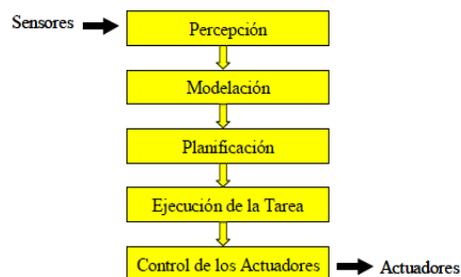


Figura 3. Proceso de control de un robot

Fuente: (Salazar, s. f.).

Las diferentes estrategias de control que pueden ser utilizadas se dividen en: (Salazar, s. f.).

- Control Reactivo (no piense, reaccione)
- Control Deliberativo (piense intensamente, luego actúe)
- Control Híbrido (Piense y actúe independientemente, en paralelo)
- Control basado en la conducta (Piense en la forma en que actúa)

1.1.5.2 Sensores

Los sensores son dispositivos físicos que miden cantidades físicas. El objetivo de los sensores es permitir que los robots puedan recibir y percibir información desde el mundo que los rodea. Su función es similar a la de nuestros sentidos (Salazar, s. f.).

El tipo de sensores a utilizar dependerá de la de propiedad física que necesita ser sensada:

Tabla 2
Algunas definiciones de robot

| PROPIEDAD FÍSICA | TECNOLOGÍA |
|----------------------|------------------------------------|
| Contacto | Switch, sensor de contacto |
| Distancia | Ultrasonido, radar, infrarrojo |
| Luz | Diodo Infrarrojo, Fotorresistencia |
| Nivel de Luz/ Imagen | Cámaras |
| Sonido | Micrófono |
| Olor | Química |
| Temperatura | Termal, infrarrojo |
| Inclinación | Inclinómetro, giroscopio |
| Altitud | Altímetro |
| Corriente | Amperímetro |

Fuente: (Salazar, s. f.).

1.1.5.3 Efectores y actuadores

Un **efector** corresponde a cualquier dispositivo que afecte o modifique al medio ambiente. Ejemplos de efectores robóticos son piernas, ruedas, brazos, dedos y pinzas. Un efector robótico esta siempre bajo el control del robot (Salazar, s. f.)...

Un **actuador** corresponde a cualquier mecanismo que permita al efector ejecutar una acción. Ejemplos de actuadores robóticos son motores eléctricos (servomotores, de paso, de corriente continua, etc.), cilindros neumáticos y cilindros hidráulicos (Salazar, s. f.).

1.1.5.4 sistema de locomoción/manipulación

Los robots pueden tener un sistema de: a) locomoción y/o de b) manipulación. El sistema de locomoción permite que el robot se mueva. El sistema de manipulación permite que el robot pueda mover o alcanzar objetos que estén cerca de él. En base a estas características los robots se dividen en robots móviles y manipuladores robóticos. (Salazar, s. f.).



Figura 4. Robot movil

Fuente: (Salazar, s. f.).



Figura 5. Robot movil

Fuente: (Salazar, s. f.).

Locomoción. A pesar de que la mayor parte de los animales utiliza piernas para desplazarse, la locomoción en base a piernas es mucho más complicada que la en base a ruedas. Esto principalmente debido a problemas de estabilidad Figura 4. (Salazar, s. f.)

Existen dos tipos de estabilidad: estática y dinámica. La estabilidad estática implica que el robot puede estar parado sin caerse. La estabilidad dinámica implica que el robot puede desplazarse sin caerse. Poseer estabilidad dinámica no quiere decir que se posea estabilidad estática (ej. saltar en un pie, equilibrar una escoba invertida en la palma de la mano). (Salazar, s. f.)

Manipulación. El objetivo de un manipulador robótico es mover parte de su cuerpo de tal forma que uno de sus efectores (ej. mano, dedo, pinza) alcance una posición y una orientación dada en el espacio. Esto con el objetivo de tomar o tocar algún objeto, figura 5. (Salazar, s. f.)

Los manipuladores robóticos más comunes son los brazos robóticos. Las **uniones** permiten conectar partes de los manipuladores. A modo de ejemplo en el caso humano tenemos la muñeca, el codo, la rodilla, etc. Los tipos de uniones más comunes son: (Salazar, s. f.)

- rotacional (rotación alrededor de un eje fijo)
- prismática (movimiento lineal)

La manipulación robótica es una tarea bastante complicada, dado que para alcanzar un objeto dado en el espacio, el robot debe poseer mecanismos de control de las uniones que permiten mover la parte del manipulador que alcanzará dicho objeto, respetando las leyes físicas de la cinemática y la dinámica. (Salazar, s. f.)

1.2 Clasificación del Robót Industrial

Un robot puede ser clasificado atendiendo a diferentes criterios o características. Algunas de éstas serán dependientes de su propia esencia, otras de la aplicación o tarea a que se destinan.

En los criterios para clasificar un robot que a continuación se expone, aparecen conceptos cuyo desarrollo será abordado en capítulos posteriores, haciéndose en este epígrafe sólo una breve reseña a su significado. (Barrientos Cruz - 2007 - Fundamentos de robótica.pdf, s. f.)

La IFR (Federación Internacional de Robótica, <http://www.ifr.org/>) clasifica los robots industriales según tres parámetros: (Isa, s. f.)

- Clasificación según el número de ejes (grados de libertad)
- Clasificación según el tipo de estructura mecánica:
- Clasificación según el tipo de control:

1.2.1 Clasificación según el número de ejes (grados de libertad)

- Robots con 3 ejes.
- Robots con 4 ejes.
- Robots con 5 o más ejes.

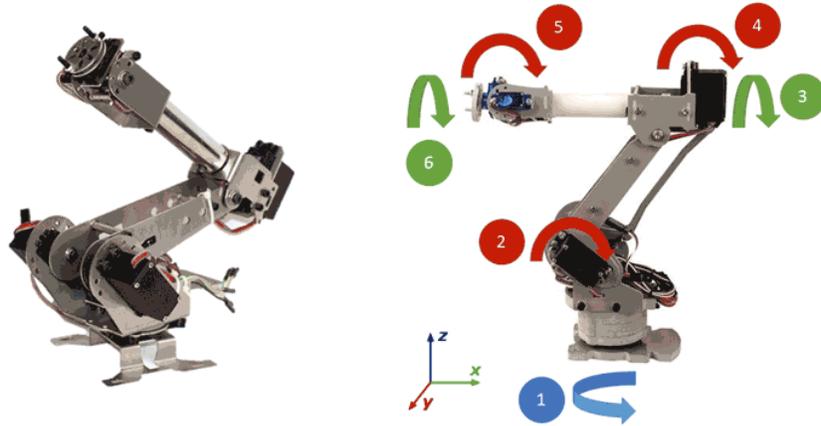


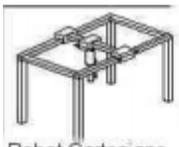
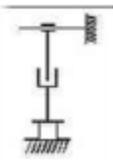
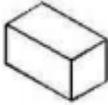
Figura 6. Robot de 6 ejes de movimiento
Fuente: («Brazo robótico controlado por Arduino», s. f.)

1.2.2 Clasificación según el tipo de estructura mecánica:

- **Cartesianos.**

Las principales características de este tipo de robots podemos observar en la tabla 3.

Tabla 3
Características de Robots Cartesianos

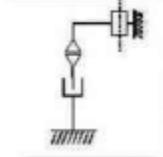
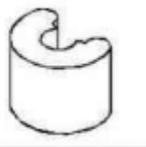
| Diagrama | Cinemática | Espacio de trabajo | Ejemplo |
|---|---|---|---|
|  <p>Robot Cartesiano</p> |  |  |  |

Fuente: (Salazar, s. f.).

- **Scara.**

Las principales características de este tipo de robots podemos observar en la tabla 4.

Tabla 4
Características de Robots Scara.

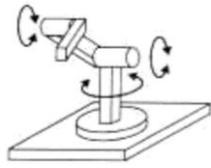
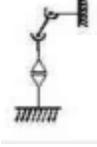
| Diagrama | Cinemática | Espacio de trabajo | Ejemplo |
|--|---|--|---|
|  <p>Robot SCARA</p> |  |  |  |

Fuente: (Salazar, s. f.).

• **Antropomórficos o angulares.**

Las principales características de este tipo de robots podemos observar en la tabla 5.

Tabla 5
Características de Robots Antropomórfico.

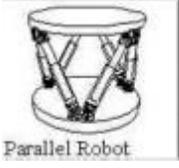
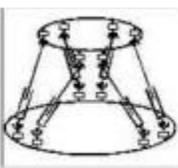
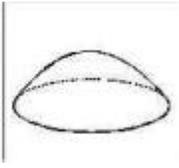
| Diagrama | Cinemática | Espacio de trabajo | Ejemplo |
|---|---|--|---|
|  |  |  |  |

Fuente: (Salazar, s. f.).

• **Paralelos**

Las principales características de este tipo de robots podemos observar en la tabla 6.

Tabla 6
Características de Robots Paralelos.

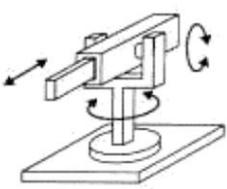
| Diagrama | Cinemática | Espacio de trabajo | Ejemplo |
|---|---|--|---|
|  |  |  |  |

Fuente: (Salazar, s. f.).

• **Esféricos.**

Las principales características de este tipo de robots podemos observar en la tabla 7.

Tabla 7
Características de Robots Esféricos.

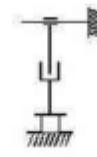
| Diagrama | Cinemática | Espacio de trabajo | Ejemplo |
|---|---|--|---|
|  |  |  |  |

Fuente: (Salazar, s. f.).

• **Cilíndricos.**

Las principales características de este tipo de robots podemos observar en la tabla 8.

Tabla 8
Características de Robots Cilíndricos.

| Diagrama | Cinemática | Espacio de trabajo | Ejemplo |
|---|---|--|---|
|  <p>Robot Cilíndrico</p> |  |  |  |

Fuente: (Salazar, s. f.).

1.2.3 Clasificación según el tipo de control

Secuencia-controlada: robot que tiene un sistema de control en el que los movimientos se realizan en un orden determinado. (Isa, s. f.)

Trayectoria-operada / continua: robot que desarrolla un procedimiento controlado donde tres o más ejes en movimiento operan de acuerdo a las especificaciones de trayectoria requerida para alcanzar la próxima posición deseada, normalmente alcanzada por interpolación. (Isa, s. f.)

Teleoperados: robot que puede ser operado de modo remoto por un operador humano. Su función es la de una extensión de las funciones del sistema motor sensorial humano. (Isa, s. f.)

Adaptativos: robot que tiene un control sensorial, adaptativo o funciones para control mediante aprendizaje. (Isa, s. f.)

- Se dice que es sensorial si el movimiento del robot y la fuerza de éste se ajusta de acuerdo a las salidas que proporcionan sensores externos.
- Se dice que el control es adaptativo si los parámetros del sistema de control son ajustados a partir de las condiciones detectadas durante el proceso.

- Se dice que es mediante aprendizaje cuando la experiencia obtenida durante ciclos anteriores se usa de modo automático para cambiar los parámetros de control y / o los algoritmos.

1.2.4 Clasificación de Robots de servicios

1. Clasificación según el tipo de interacción: (Isa, s. f.)

- Servicio destinado a seres humanos: personal, seguridad, entretenimiento etc.
- Servicio destinado a equipamiento: mantenimiento, reparación, limpieza, etc.
- Otro tipo de servicios: transporte, adquisición de datos y todos aquellos que no se puedan clasificar como servicio destinado a seres humanos o equipamiento.



Figura 7. Robot de servicios tipo de interacción

Fuente: («Los robots facilitan el trabajo de los humanos cada vez más», 2013)

2. Clasificación según áreas de aplicación: (Isa, s. f.)

- De limpieza: limpieza de ventanas, muros, tanques, suelos, etc.
- De alcantarillado: destinados a limpieza e inspección.
- Caminantes y escaladores: destinados a limpieza e inspección
- De inspección de plantas industriales, centrales nucleares, puentes, etc.
- Submarinos: destinados a todo tipo de trabajo bajo el agua.

- Domésticos: destinados a labores dentro de las casas particulares.
- Médicos: destinados a labores médicas, operaciones quirúrgicas, etc.
- De asistencia: destinados para la ayuda a personas discapacitadas (silla de ruedas robotizada).



Figura 8. Robot de servicios tipo de interacción
Fuente: (*Robots de servicios seguridad*, s. f.)

1.3 Estructura mecánica del robot.

Mecánicamente, un robot está formado por una serie de elementos o eslabones unidos mediante articulaciones que permiten un movimiento relativo entre cada dos eslabones consecutivos. La constitución física de la mayor parte de los robots manipuladores industriales guarda cierta similitud con la anatomía del brazo humano, por lo que, en ocasiones, para hacer referencia a los distintos elementos que componen el robot, se usan términos como cuerpo, brazo, codo y muñeca. (Reyes & Rodríguez, 2013)

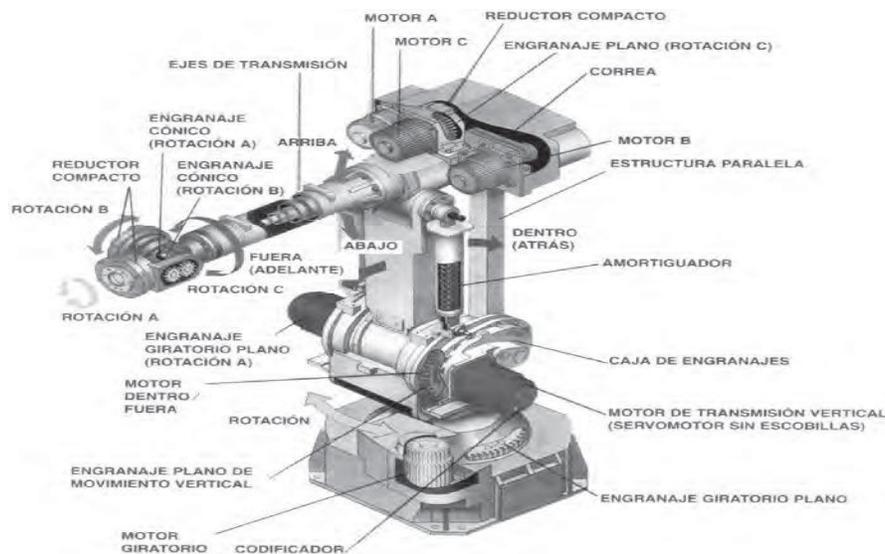


Figura 9. Ejemplo de estructura mecánica y elementos constitutivos de un robot.
Fuente: (Reyes & Rodríguez, 2013)

El movimiento de cada articulación puede ser de desplazamiento, de giro, o una combinación de ambos. En 1876 Franz Reuleaux [REULEAUX-1876] identificó los posibles movimientos relativos entre dos elementos en contacto. Reuleaux denominó pares inferiores (*lower pairs*) a aquellos en los que el contacto se realiza entre superficies, mientras que si el contacto es puntual o lineal, los denominó pares superiores. La Figura 10 muestra los seis posibles pares inferiores establecidas por Reuleaux. (Reyes & Rodríguez, 2013).

1.3.1 Grados de libertad

Cada uno de los movimientos independientes que puede realizar cada articulación con respecto a la anterior, se denomina grado de libertad (GDL). En la Figura 10 se indica el número de GDL de cada tipo de articulación correspondiente a los pares inferiores. Como se observa en ella, sólo tres tipos de articulaciones tienen un solo grado de libertad (éstos se denominan pares lineales o de primer grado). En la práctica, en robótica sólo se emplean las articulaciones de rotación y prismática. En caso de que un robot tuviera alguna articulación con más de un grado de libertad, se podría asumir que se trata de varias articulaciones diferentes, unidas por eslabones de longitud nula. Una cadena cinemática, es una serie de eslabones o barras unidas por articulaciones. La estructura mecánica de un robot manipulador constituye una cadena cinemática. (Reyes & Rodríguez, 2013).

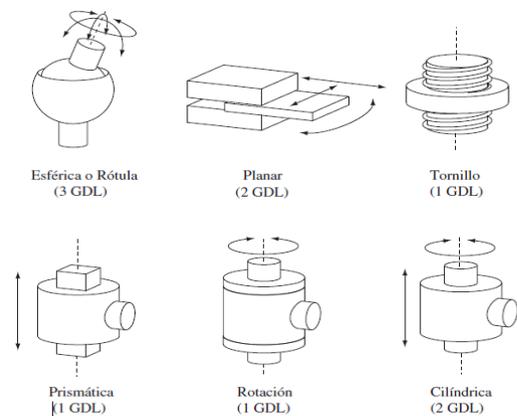


Figura 10. Los seis pares inferiores de Reuleaux.
Fuente:(Reyes & Rodríguez, 2013)

Cuando en una cadena cinemática se puede llegar desde cualquier eslabón a cualquier otro mediante al menos dos caminos, se dice que se trata de una cadena cinemática cerrada. En caso de que sólo haya un camino posible se dirá que se trata de una cadena cinemática abierta (Véase Figura 11). (Reyes & Rodríguez, 2013).

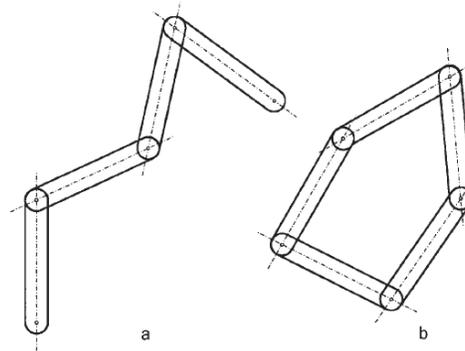


Figura 2.3. Cadena cinemática: a) abierta y b) cerrada.

Figura 11. Cadena cinemática: a) abierta y b) cerrada..
Fuente:(Reyes & Rodríguez, 2013)

El número de grados de libertad de una cadena cinemática puede ser obtenido mediante la fórmula de Grübler, según la cual:

$$NGDL = \lambda \cdot (n - j - 1) + \sum_{i=1}^j f_i$$

Donde:

λ : GDL del espacio de trabajo (Típicamente tres en el plano, seis en el espacio).

n : Número de eslabones (debe incluirse el eslabón fijo o base).

j : número de articulaciones.

f_i : Grados de libertad permitidos a la articulación i .

Así, en las cadenas mostradas en la Figura 11, se tendrán los valores mostrados en la Tabla 9 (Reyes & Rodríguez, 2013)..

Tabla 9
Aplicación fórmula de Grübler

| | Figura 2.3a | Figura 2.3b |
|-----------|-----------------------------|---|
| λ | 3 | 3 |
| n | 4 | 5 |
| j | 3 | 5 |
| f_i | $f_1 = 1; f_2 = 1; f_3 = 1$ | $f_1 = 1; f_2 = 1; f_3 = 1; f_4 = 1; f_5 = 1$ |
| NGDL | 3 | 2 |

Fuente:(Reyes & Rodríguez, 2013)

Los robots manipuladores son, en la mayor parte de los casos, cadenas cinemáticas abiertas con las articulaciones de tipo rotación o prismática (con un solo GDL cada una), siendo por lo general sencillo encontrar el número de grados de libertad del robot, pues coincide con el número de articulaciones de que se compone. No obstante, en los robots con cadena cinemática cerrada, que pueden usar otro tipo de articulaciones, como las esféricas, es preciso evaluar con mayor rigor el número de GDL. (Reyes & Rodríguez, 2013).

1.3.2 Practica de mecanismos con SAM 6.0 - The Ultimate Mechanism Designer

Para entender sobre los mecanismos de un robot se ve en la necesidad de instalar un software que ayude con la comprensión de los sistemas mecánicos y entender de esa manera tanto la comprensión de grados de libertad como articulaciones y eslabones.

Antes que nada, utilizaremos el software SAM 6.0. A continuación se puede ver los siguientes pasos para su instalación:

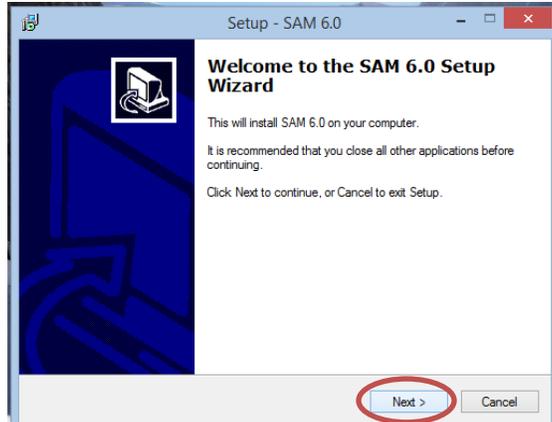


Figura 12. Pasos de instalaciones para Sam 6.0 num1
Fuente: Autores

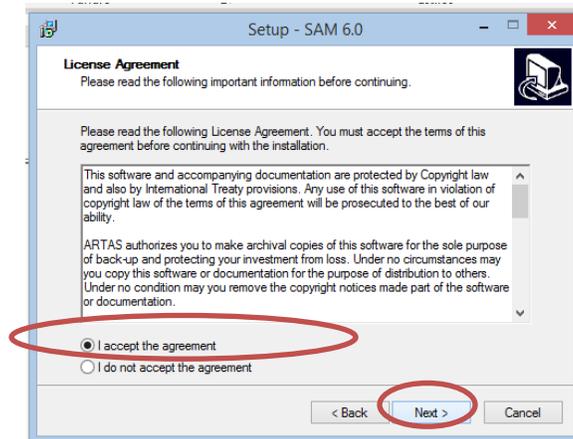


Figura 13. Pasos de instalaciones para Sam 6.0 num2
Fuente: Autores (Reyes Cortés & Rodríguez Cruz, 2013)

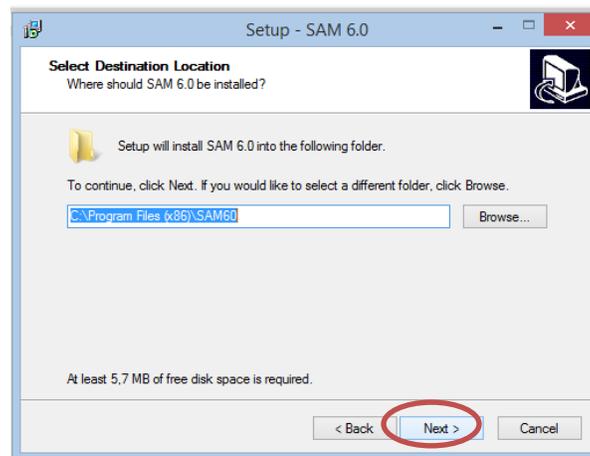


Figura 14. Pasos de instalaciones para Sam 6.0 num3
Fuente: Autores

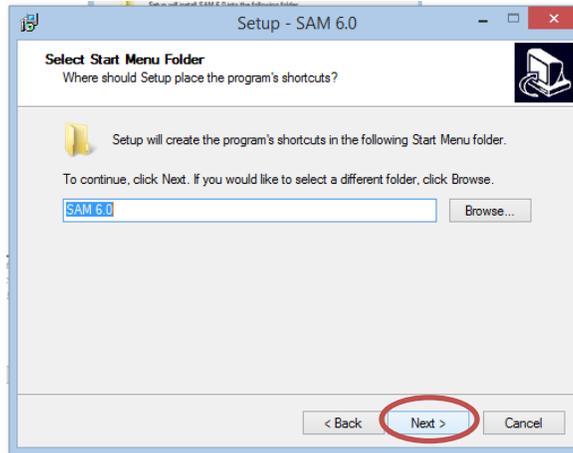


Figura 15. Pasos de instalaciones para Sam 6.0 num4
Fuente: Autores

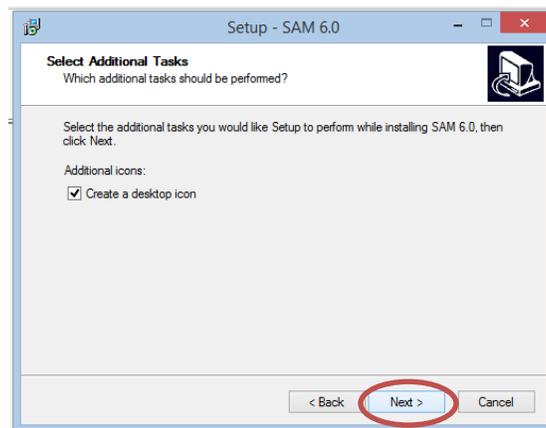


Figura 16. Pasos de instalaciones para Sam 6.0 num5
Fuente: Autores

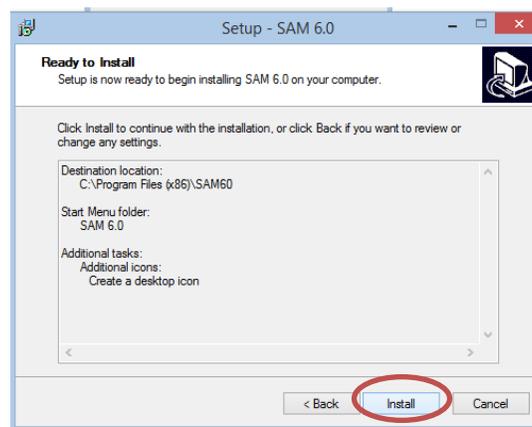


Figura 17. Pasos de instalaciones para Sam 6.0 num6
Fuente: Autores

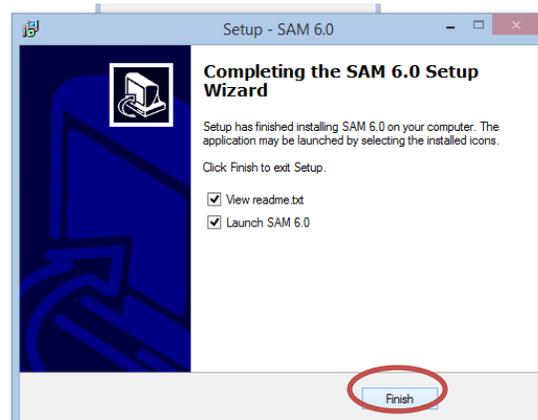


Figura 18. Pasos de instalaciones para Sam 6.0 num7
Fuente: Autores

Ya que este software es una versión de prueba o Trail se pide cada vez que se necesite realizar un ejercicio instalarlo. Después de haber instalado el software se tendrá la siguiente pantalla la misma que deberemos escoger la opción por defecto TRIAL.

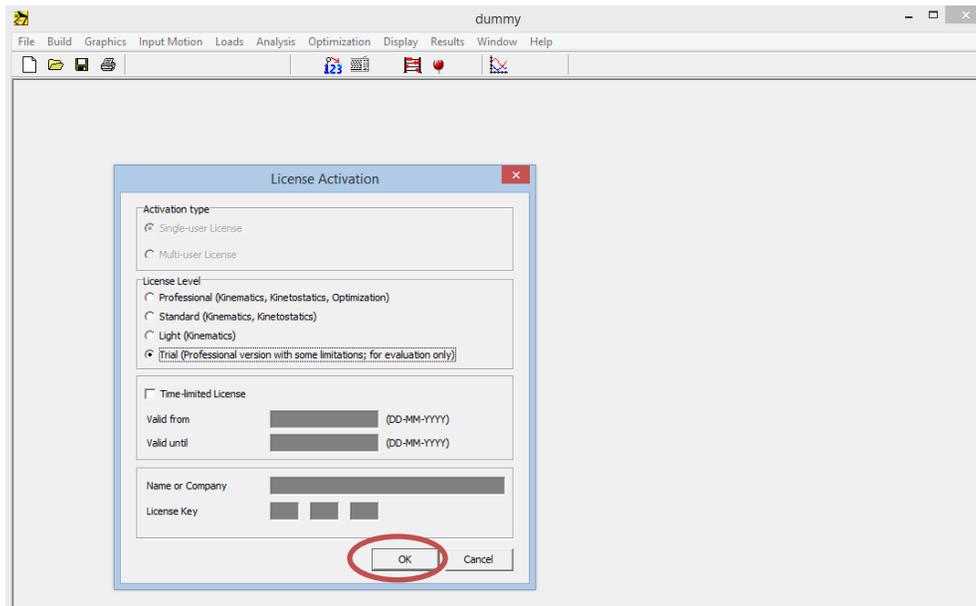


Figura 19. Pasos de instalaciones para Sam 6.0 prueba de inicio
Fuente: Autores

Para la explicación del primer ejercicio se deja el siguiente link para realizar las pruebas de mecanismos correspondientes.

<https://www.youtube.com/watch?v=Pi18aYBg5I4>

Con esta práctica estaremos preparados para realizar los ejemplos de brazos con mecanismo de 3 y 4 GDL

1.4 Herramientas matemáticas para localización espacial

Las herramientas matemáticas para la robótica son de gran ayuda para establecer un posicionamiento de un robot ya sea en un plano o en el espacio, también estas ayudaran a realizar un modelamiento previo del sistema robótico antes de ponerlo en funcionamiento ayudando a que se maneje temas de seguridad tanto para la persona que opera y programa una estructura robótica como también para el robot.

La herramientas matemáticas utilizadas para procesos de brazos robots son sistemas vectoriales y marices del algebra lineal, estas mediante el manejo de operaciones básicas como traslación, traspuesta, rotación , multiplicación ,suma, resta de matrices ayudaran a manejar y a entender la rotación y el posicionamiento de un robot. Par facilitar la explicación del tema dejo el siguiente link para poder profundizar en el tema:

<https://www.youtube.com/watch?v=LvAd8jwgEaA>

1.4.1 Introducción

La manipulación de la pieza llevada a cabo por el robot implica el movimiento espacial de su extremo.

Para que el robot pueda manipular una pieza, es necesario conocer su LOCALIZACIÓN, es decir la posición y orientación de ésta con respecto a la base del robot. (Legarreta & Martinez, s. f.)

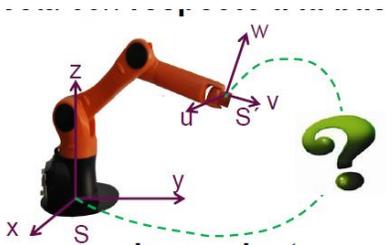


Figura 21. Posición y orientación respecto a la base de un robot
Fuente:(Legarreta & Martinez, s. f.)

Necesidad de una herramienta matemática para especificar la posición y orientación del extremo del robot respecto la base del robot.

El Sistema de coordenadas de **referencia** situado en la base del robot se denomina **{S}** y sus **ejes** asociados **XYZ**, formando el sistema OXYZ.

El Sistema de coordenadas situado en la muñeca del robot se denomina **{S'}** y sus **ejes** asociados **UVW**, formando el sistema O'UVW

1.4.2 Localización Espacial

La localización espacial se puede realizar en base a la representación de:

- Posición
 - Coordenadas cartesianas.
 - Coordenadas cilíndricas.
 - Coordenadas esféricas
- Orientación.
 - Matrices de rotación.
 - Ángulos de Euler.
 - Par de rotación.
- Localización (Posición + Orientación)
 - Coordenadas homogéneas.
 - Matrices de transformación homogénea (MTH).

Se puede posicionar un punto en el plano o en el espacio.

- **Posición en el PLANO**



Figura 22. Posición en el plano
Fuente:(Legarreta & Martinez, s. f.)

- **Posición en el ESPACIO**

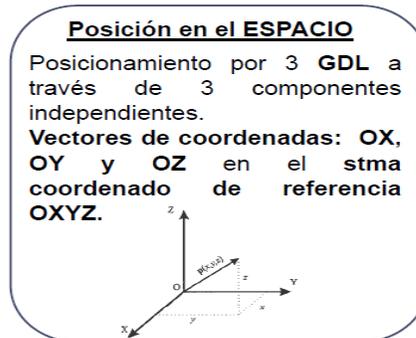
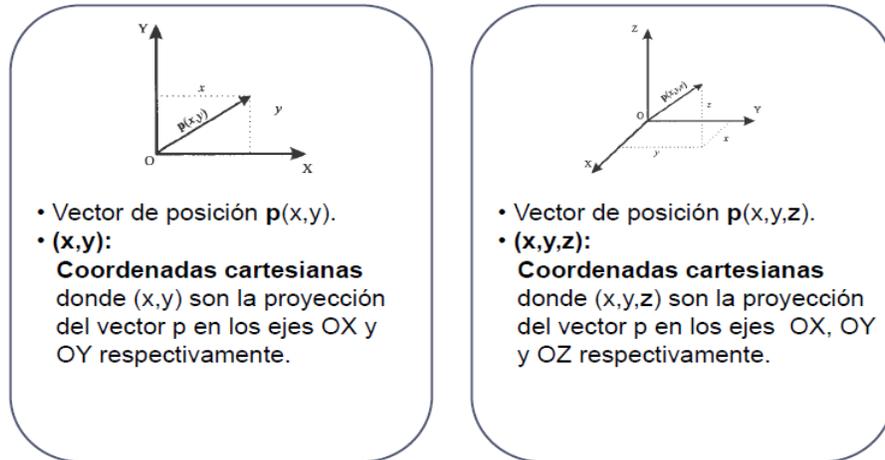


Figura 23. Posición en el espacio
Fuente:(Legarreta & Martinez, s. f.)

Existen diferentes sistemas de coordenadas para posicionar un punto, estas son: cartesianas, polares, cilíndricas y esféricas.

1.4.2.1 Posición Coordenadas cartesianas.

En base a su posición en el plano o el espacio se puede ver en la siguiente figura 24



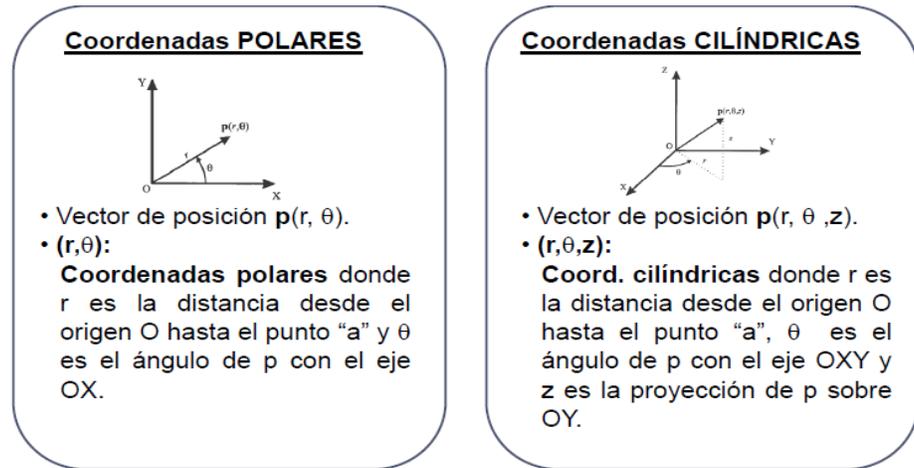
- Vector de posición $p(x,y)$.
- (x,y) :
Coordenadas cartesianas donde (x,y) son la proyección del vector p en los ejes OX y OY respectivamente.

- Vector de posición $p(x,y,z)$.
- (x,y,z) :
Coordenadas cartesianas donde (x,y,z) son la proyección del vector p en los ejes OX , OY y OZ respectivamente.

Figura 24. Coordenadas cartesianas.
Fuente:(Legarreta & Martinez, s. f.)

1.4.2.2 Posición Coordenadas Polares y Cilíndricas.

Se puede representar dichas coordenadas en base a la siguiente figura.



- Coordenadas POLARES**
- Vector de posición $p(r, \theta)$.
 - (r,θ) :
Coordenadas polares donde r es la distancia desde el origen O hasta el punto "a" y θ es el ángulo de p con el eje OX .

- Coordenadas CILÍNDRICAS**
- Vector de posición $p(r, \theta, z)$.
 - (r,θ,z) :
Coord. cilíndricas donde r es la distancia desde el origen O hasta el punto "a", θ es el ángulo de p con el eje OXY y z es la proyección de p sobre OY .

Figura 25. Coordenadas Polares y Cilíndricas.
Fuente:(Legarreta & Martinez, s. f.)

1.4.2.3 Posición Coordenadas esféricas.

La posición mediante coordenadas esféricas se puede representar con las siguientes variables como se muestra en la figura 26.

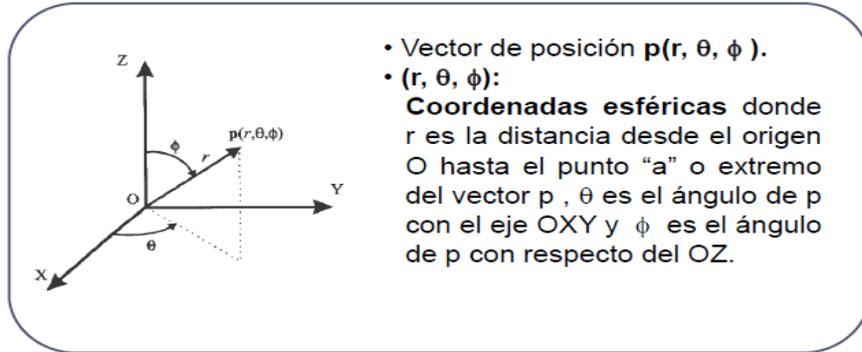


Figura 26. Coordenadas Polares y Cilíndricas.

Fuente:(Legarreta & Martinez, s. f.).

1.4.2.4 Conversiones y Ejemplos

Conversiones coordenadas cartesianas (x,y) a polares (r,θ) son:

$$r = \sqrt{x^2 + y^2}$$

$$\theta = \text{atan}(y / x)$$

$$x = r \times \cos(\theta)$$

$$y = r \times \sin(\theta)$$

Conversión de coordenadas cilíndricas a rectangulares

$$x = r \cos \theta$$

$$y = r \sin \theta$$

$$z = z$$

Conversión de coordenadas rectangulares a cilíndricas

$$r^2 = x^2 + y^2$$

$$\tan \theta = \frac{y}{x}$$

$$z = z$$

Conversión de coordenadas esféricas a rectangulares

$$x = \rho \sin \phi \cos \theta$$

$$y = \rho \sin \phi \sin \theta$$

$$z = \rho \cos \phi$$

Conversión de coordenadas rectangulares a esféricas

$$\rho^2 = x^2 + y^2 + z^2$$

$$\tan \theta = \frac{y}{x}$$

$$\phi = \cos^{-1}\left(\frac{z}{\sqrt{x^2 + y^2 + z^2}}\right)$$

Conversión de coordenadas esféricas a cilíndricas ($r \geq 0$)

$$r^2 = \rho^2 \sin^2 \phi$$

$$\theta = \theta$$

$$z = \rho \cos \phi$$

Conversión de coordenadas cilíndricas a esféricas ($r \geq 0$)

$$\rho = \sqrt{r^2 + z^2}$$

$$\theta = \theta$$

$$\phi = \cos^{-1}\left(\frac{z}{\sqrt{r^2 + z^2}}\right)$$

Problemas resueltos.

Problema 1. Convertir el punto $(r, \theta, z) = (4, \frac{5\pi}{6}, 3)$ a coordenadas rectangulares.

Solución. Se llevará a cabo la conversión de coordenadas cilíndricas a rectangulares. Para ello, se sabe que, $r = 4$, $\frac{5\pi}{6}$ y $z = 3$, entonces, utilizando las fórmulas y sustituyendo

$$x = r \cos \theta$$

$$x = 4 \cos \frac{5\pi}{6} = 4\left(-\frac{\sqrt{3}}{2}\right)$$

$$x = -2\sqrt{3}$$

$$y = r \sin \theta$$

$$y = 4 \sin \frac{5\pi}{6} = 4\left(\frac{1}{2}\right)$$

$$y = 2$$

$$z = z$$

$$z = 3$$

Finalmente, las coordenadas rectangulares son $(-2\sqrt{3}, 2, 3)$

Problema 2. Convertir el punto $(x, y, z) = (1, \sqrt{3}, 2)$ a coordenadas cilíndricas.

Solución. Se llevará a cabo la conversión de coordenadas rectangulares a cilíndricas. Para ello, se sabe que, $x = 1$, $y = \sqrt{3}$ y $z = 2$, entonces, utilizando las fórmulas y sustituyendo

$$r^2 = x^2 + y^2$$

$$r = \sqrt{x^2 + y^2}$$

$$r = \sqrt{(1)^2 + (\sqrt{3})^2}$$

$$r = \sqrt{1 + 3} = \sqrt{4}$$

$$r = 2$$

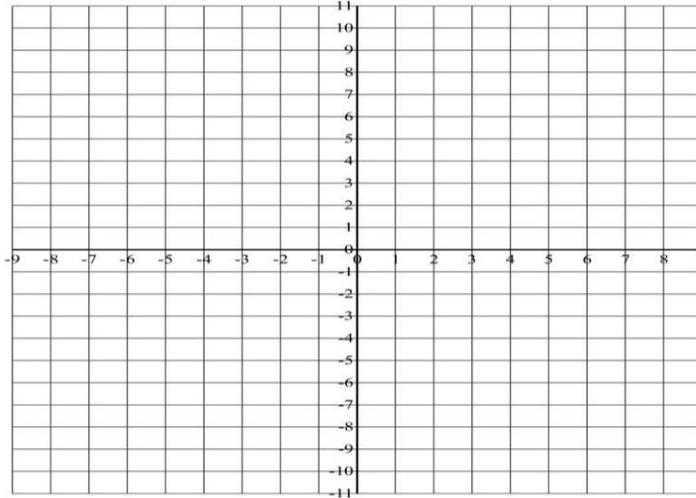
ACTIVIDADES UNIDAD 1

Actividad 1: Preguntas

1. ¿ En qué fecha se realizó el primer sistema de telemanipulación con servocontrol bilateral?
2. ¿Cuál es la definición de Robót según la Enciclopedia Británica?
3. ¿Qué es un Robót industrial manipulador?
4. ¿En que se resumen las leyes de la Robótica?
5. ¿De qué consta la estructura interna de un Robót?
6. ¿Cuál es la diferencia entre efector y actuador?
7. ¿Cuáles serían ejemplos de manipuladores?
8. Según la Federación Internacional de Robótica, ¿Como se clasifican a los robots industriales?
9. ¿De que consta principalmente la Estructura mecánica de un robot?
10. ¿Qué se entiende por grados de libertad o GDL?
11. ¿Cuál es la definición de eslabón?
12. ¿Cuál es la definición de articulación?
13. ¿Para qué se utilizan las herramientas matemáticas para localización espacial?
14. ¿Qué tipo de sistemas coordenadas se utilizan para posicionar un punto en el espacio?
15. ¿Cuáles son las coordenadas cilíndricas para determinar una posición?

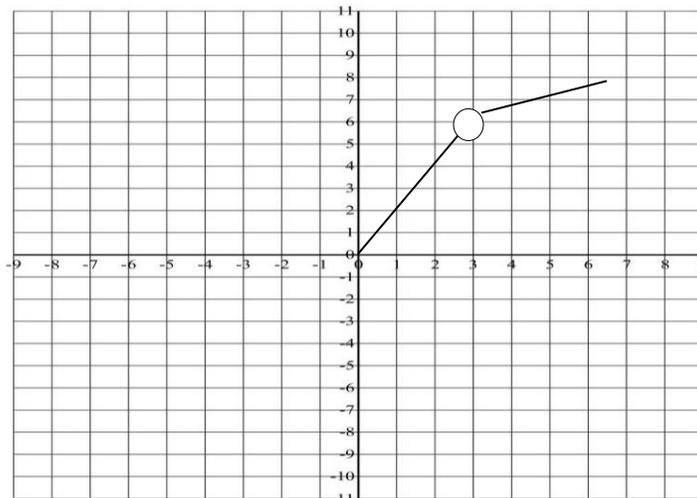
Actividad 2: Ejercicios

1.- Se tiene una articulación mecánica de 1,7 m el cual se puede mover un ángulo de elevación de 30° , cuál sería su posición en base a coordenadas rectangulares sen el plano.



Fuente: (Autores)

2.- Dado la figura cinemática de la siguiente figura se pide obtener las coordenadas polares de la articulación base , sabiendo que desde la pared vertical máximo llega puede llegar al articulación 20°

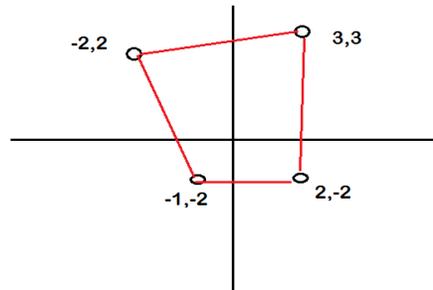


Fuente: (Autores)

3.- Realizar la conversión de las coordenadas generadas en el primer ejercicio a coordenadas cilíndricas

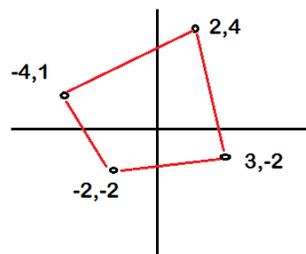
Actividad 3: Actividades de simulación/ prácticas.

1.- Realizar la simulación del siguiente mecanismo de doble manibela y observar cuantos grados de libertad existen utilizar, sam 6.0 para la implementación y simulación.



Fuente: (Autores)

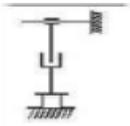
2.- Dado el siguiente sistema impleméntelo en sam 6.0 e identifique que acción realiza y cual es la diferencia del anterior sistema mecánico.



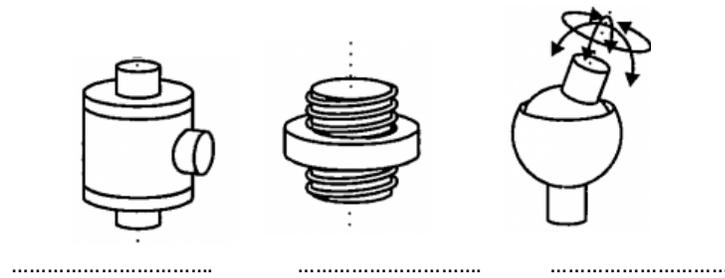
Fuente: (Autores)

Actividad 4: Autoevaluación

- 1.- En similitudes cuantos grados de libertad tiene un robot que simula el proceso de articulaciones superior del ser humano que van desde la cintura hasta los dedos y cuales son. (realice un dibujo si es preciso).
- 2.- De que consta un sub sistema de un robot manipulador?
- 3.- Según su cinemática y estructura mecánica que tipo de robot perteneces cada uno

| Cinemática | Tipo de robot |
|---|---------------|
|  | |
|  | |

4.- Cuál es el número de componentes de movimiento simples que se necesitan para generar un movimiento complejo de cada componente mecánico?



5.- La función de este sistema es detectar las acciones que ejecuta el robot de tal forma que pueda cumplir con la tarea que le ha sido asignada?

Verdad

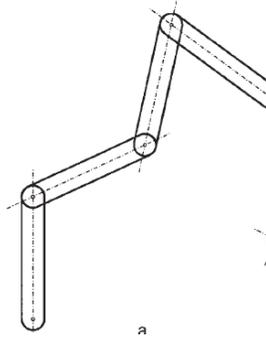
Falso

Por que?:.....

6.- Según la propiedad física de ALTITUD el dispositivo tecnológico con la que se puede medir esta es:

- a) Altimetro
- b) Multímetro
- c) Amperímetro.

7.- Aplicando la fórmula de Grübler obtener el número de grados de libertad de la siguiente imagen



Fuente: (Autores)

8.- Para que se utilizan las herramientas matemáticas en robótica?

- a) especificar la posición y localización en el plano
- b) especificar la posición y velocidad
- c) especificar la posición y orientación

9.- Completar: La localización espacial se puede realizar en base a la representación de:

.....
.....
.....
.....

10.- Dadas las coordenadas rectangulares de la posición inicial de robot (x,y) donde $x=25$ cm y $y=6$ = 56 pulgadas. Se pide encontrar la representación de las mismas en coordenadas polares.

UNIDAD 2: SENSORES Y ACTUADORES

2.1 Transmisores y reductores

2.1.1 Transmisores Mecánicos para robótica

Dado que un robot mueve su extremo con aceleraciones elevadas, es de gran importancia reducir al máximo su momento de inercia. Del mismo modo, los pares estáticos que deben vencer los actuadores dependen directamente de la distancia de las masas al actuador. Por estos motivos se procura que los actuadores, por lo general pesados, estén lo más cerca posible de la base del robot. Esta circunstancia obliga a utilizar sistemas de transmisión que trasladen el movimiento hasta las articulaciones, especialmente a las situadas en el extremo del robot. Asimismo, las transmisiones pueden ser utilizadas para convertir movimiento circular en lineal o viceversa, cuando sea necesario (González, 2002b).

Existen actualmente en el mercado robots industriales con acoplamiento directo entre accionamiento y articulación. Se trata, sin embargo, de casos particulares dentro de la generalidad que en los robots industriales actuales supone la existencia de sistemas de transmisión junto con reductores para el acoplamiento entre actuadores y articulaciones (González, 2002b).

Es de esperar que un buen sistema de transmisión cumpla con una serie de características básicas: (González, 2002b)

- debe tener un tamaño y peso reducido;
- se ha de evitar que presente juegos u holguras considerables;
- se deben buscar transmisiones con gran rendimiento.

Aunque no existe un sistema de transmisión específico para los robots, sí existen algunos usados con mayor frecuencia, y que se mencionan en la tabla siguiente. La clasificación se ha realizado sobre la base del tipo de movimiento posible en la entrada y la salida: lineal o circular. (González, 2002b)

Tabla 10
Sistemas de transmisión para robots

| Sistemas de transmisión para robots | | | |
|--|--------------------------|---|--|
| Entrada-Salida | Denominación | Ventajas | Inconvenientes |
| Circular-Circular | Engranaje | Pares altos Distancia grande Distancia grande | Holguras Ruido Giro limitado Deformabilidad |
| | Correa dentada | | |
| | Cadena | | |
| | Paralelogramo | | |
| | Cable | | |
| Circular-Lineal | Tornillo sin fin | Poca holgura | Rozamiento |
| | Cremallera | Holgura media | |
| Lineal-Circular | Paralelogramo articulado | Holgura media | Control difícil Rozamiento |
| | Cremallera | | |
| | | | |

Fuente: (González, 2002b)

En esta tabla también quedan reflejadas algunas ventajas e inconvenientes propios de algunos sistemas de transmisión. Entre ellas cabe destacar la holgura o juego. Es muy importante que el sistema de transmisión a utilizar no afecte al movimiento que transmite, ya sea por el rozamiento inherente a su funcionamiento o por las holguras que su desgaste pueda introducir. También hay que tener en cuenta que el sistema de transmisión sea capaz de soportar un funcionamiento continuo a un par elevado y, a ser posible, entre grandes distancias (González, 2002b).

Las transmisiones más habituales son aquellas que cuentan con movimiento circular tanto a la entrada como a la salida. Incluidas en éstas se encuentran los engranajes, las correas dentadas y las cadenas (González, 2002b).

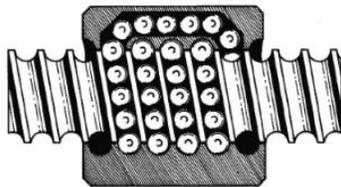
En base a esto se puede decir entonces que:

Los trasmisores son: “Elementos encargados de transmitir el movimiento desde los actuadores hasta las articulaciones (Fraile, s. f.).

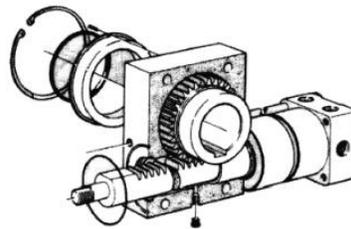
- El robot se mueve con aceleraciones elevadas \Rightarrow es necesario reducir al máximo su momento de inercia.
- Por ello, los actuadores deben estar cerca de la base del robot \Rightarrow es necesario utilizar sistemas de transmisión que trasladen el movimiento desde el actuador hasta la articulación del robot.
- Transforman movimiento Circular en lineal o viceversa.

2.1.1.1 Conversión del movimiento circular a lineal:

En la siguiente figura podemos ver transmisores mecánicos para realizar la conversión del movimiento circular a lineal.



Tornillo sin fin de circulación de bolas



Conjunto piñón-cremallera

Figura 27. Conversión del movimiento circular a lineal.

Fuente: (Fraile, s. f.)

2.1.1.2 Conversión del movimiento lineal a circular:

En la siguiente figura podemos ver transmisores mecánicos para realizar la conversión del movimiento lineal a circular.

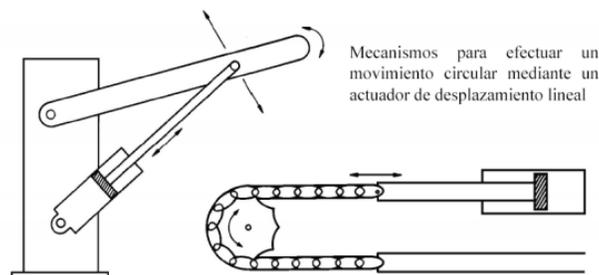


Figura 28. Conversión del movimiento lineal a circular.

Fuente: (Fraile, s. f.)

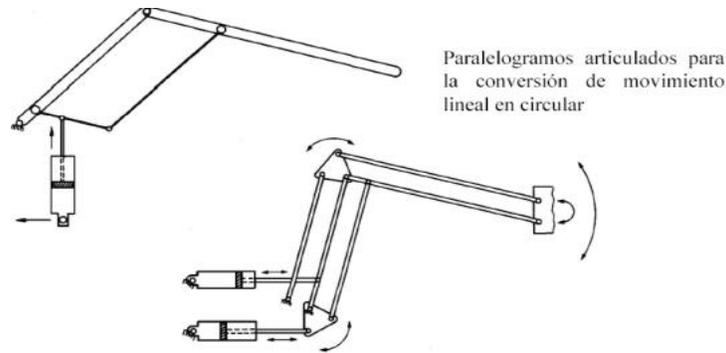


Figura 29. Conversión del movimiento lineal a circular por paralelogramos.
Fuente: (Fraile, s. f.)

2.1.2 Reductores Mecánicos para robótica

En cuanto a los reductores, al contrario que con las transmisiones, sí existen determinados sistemas usados de manera preferente en los robots industriales. Esto se debe a que a los reductores utilizados en robótica se les exigen unas condiciones de funcionamiento muy restrictivas. La exigencia de estas características viene motivada por las altas prestaciones que se le piden al robot en cuanto a precisión y velocidad de posicionamiento. La siguiente tabla muestra valores típicos de los reductores para robótica actualmente empleados. (González, 2002b)

Tabla 11
Valores típicos para reductores en robótica

| Características de los reductores para robótica | |
|---|--------------------------|
| Características | Valores típicos |
| Relación de reducción | 50 / 300 |
| Peso y tamaño | 0.1 / 30 kg |
| Momento de inercia | 0.0001 kg m ² |
| Velocidades de entrada máxima | 6000 / 7000 rpm |
| Par de salida nominal | 5700 N m |
| Par de salida máximo | 7900 N m |
| Juego angular | 0-2" |
| Rigidez torsional | 100 / 2000 N m/rad |
| Rendimiento | 85% / 98% |

Fuente: (González, 2002b)

Se buscan reductores de bajo peso, reducido tamaño, bajo rozamiento y que al mismo tiempo sean capaces de realizar una reducción elevada de velocidad en un único paso. Se tiende también a minimizar su momento de inercia, de negativa influencia en el funcionamiento del motor, especialmente crítico en el caso de motores de baja inercia (González, 2002b).

Los reductores, por motivos de diseño, tienen una velocidad máxima admisible, que como regla general aumenta a medida que disminuye el tamaño del motor. También existe una limitación en cuanto al par de entrada nominal permisible (T_2) que depende del par de entrada (T_1) y de la relación de transmisión a través de la relación:

$$T_2 = \eta T_1 (w_1 / w_2)$$

Donde el rendimiento, η , puede llegar a ser cerca del 100%, y la relación de reducción de velocidades (w_1 = velocidad de entrada; w_2 = velocidad de salida) varía entre 50 y 300.

Puesto que los robots trabajan en ciclos cortos, que implican continuos arranques y paradas, es de gran importancia que el reductor sea capaz de soportar pares elevados puntuales. También se busca que el juego angular sea lo menor posible. Éste se define como el ángulo que gira al eje de salida cuando se cambia su sentido de giro sin que llegue a girar al eje de entrada (González, 2002b).

Por último, es importante que los reductores para robótica posean una rigidez torsional, definida como el par que hay que aplicar sobre el eje de salida para que, manteniendo bloqueado el de entrada, aquél gire un ángulo unidad (González, 2002b).

Podemos decir entonces de un Reductor: “Su misión es adaptar el par y la velocidad de la salida del actuador a los valores adecuados para el movimiento de los brazos del actuador”

- Bajo peso, tamaño reducido, bajo rozamiento, elevada capacidad de reducción y bajo momento de inercia. Objetivo: lograr elevada precisión y buena velocidad de posicionamiento del robot.
- Como el robot trabaja en ciclos cortos \Rightarrow los reductores deben soportar pares elevados.
- Juego angular bajo. Este es el ángulo que gira el eje de salida cuando se cambia su sentido de giro sin que llegue a girar el eje de entrada.

- Alta rigidez torsional, definida como el par que hay que aplicar sobre el eje de salida para que, manteniendo bloqueada el de la entrada, aquel gire un ángulo unitario.
- Reductor mas conocido: “Harmonic Drive ”

2.1.2.1 Reductor Harmonic Drive:

Los engranajes de “harmonic drive” tienen 3 componentes principales: (Fraile, s. f.)

- Un splín circular (“circular spline”): Es una corona exterior rigida con dentado interior.
- Un anillo flexible (“flexspline”): Es un vaso flexible con dentado exterior que engrana con el “circular spline”, y normalmente está conectado al eje de salida.
- Un generador de onda (“wave generator”): Es un rodamiento elipsoidal al que está conectado al eje de entrada.

2.1.2.2 Estructura Reductor Harmonic Drive:

Para mostrar la estructura y de que esta compuesto el reductor tipo Harmonic Drive se puede ver en la siguiente figura.

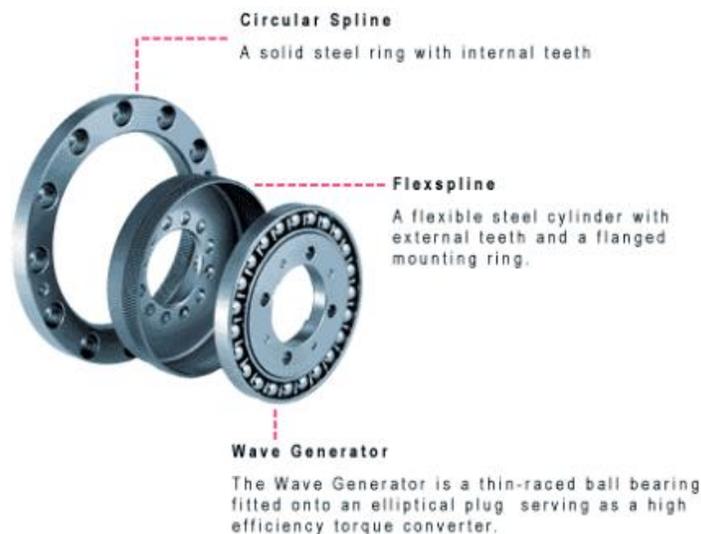


Figura 30. Estructura y elementos de un Reductor Harmonic Drive
Fuente: (Fraile, s. f.)

2.1.2.3 Características Reductor Harmonic Drive:

Dentro de las principales características se puede mencionar: (Fraile, s. f.)

- El Circular Spline es rígido y está fijo.
- El Flexspline es flexible y se une al eje de salida.
- El Wave Generator es elíptico y se fija al eje de entrada.
- Los dientes interiores del Circular Spline engranan con los exteriores del Flexspline.
- La diferencia de dientes es de 1 o 2.
- La relación de reducción es la relación entre dicha diferencia y los dientes del Flexspline.

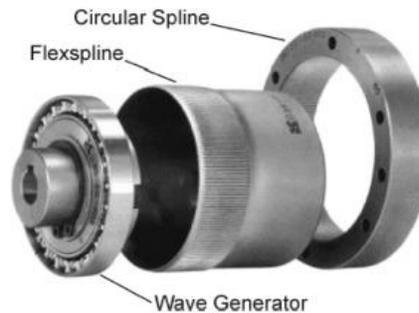


Figura 31. Características de un Reductor Harmonic Drive
Fuente: (Fraile, s. f.)

2.2 Actuadores (Neumáticos, hidráulicos y eléctricos)

Los actuadores tienen como misión generar el movimiento de los elementos del robot según las ordenes dadas por la unidad de control. Se clasifican en tres grandes grupos, según la energía que utilizan: (González, 2002b).

- Neumáticos
- Hidráulicos
- Eléctricos

Los actuadores neumáticos utilizan el aire comprimido como fuente de energía y son muy indicados en el control de movimientos rápidos, pero de precisión limitada. Los motores hidráulicos son recomendables en los manipuladores que tienen una gran capacidad de carga, junto a una precisa regulación de velocidad. Los motores eléctricos son los más utilizados, por su fácil y preciso control, así como por otras propiedades ventajosas que establece su funcionamiento, como consecuencia del empleo de la energía eléctrica. Más tarde se proporcionará una

comparación detallada entre los diferentes tipos de actuadores utilizados en robótica (González, 2002b).

Cada uno de estos sistemas presenta características diferentes, siendo preciso evaluarlas a la hora de seleccionar el tipo de actuador más conveniente. Las características a considerar son, entre otras:

- Potencia
- Controlabilidad
- Peso y volumen
- Precisión
- Velocidad
- Mantenimiento
- Coste

2.2.1 Actuadores neumáticos

En ellos la fuente de energía es aire a presión entre 5 y 10 bar. Existen dos tipos de actuadores neumáticos:

- Cilindros neumáticos
- Motores neumáticos (de aletas rotativas o de pistones axiales).

2.2.1.1 Cilindros neumáticos

En estos se consigue el desplazamiento de un émbolo encerrado en un cilindro como consecuencia de la diferencia de presión a ambos lados de aquél. Los cilindros neumáticos pueden ser *de simple o de doble efecto*. En los primeros, el émbolo se desplaza en un sentido como resultado del empuje ejercido por el aire a presión, mientras que en el otro sentido se desplaza como consecuencia del efecto de un muelle (que recupera al émbolo a su posición en reposo) (González, 2002b).

En los cilindros de doble efecto el aire a presión es el encargado de empujar al émbolo en las dos direcciones, al poder ser introducido de forma arbitraria en cualquiera de las dos cámaras.

En la figura 32 se puede observar cómo está constituido un cilindro de simple efecto con su símbolo y en la figura 33 la constitución de un cilindro doble efecto con su símbolo

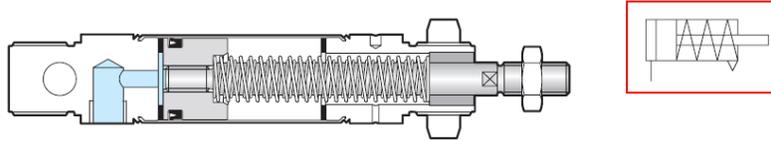


Figura 32. Cilindro de simple efecto – dibujo en sección y símbolo
Fuente: (Ebel & Idler, 2019)

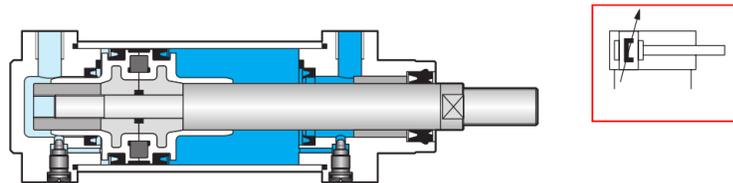


Fig. 4.6: Cilindro de doble efecto con amortiguación en las posiciones finales – Dibujo en sección y símbolo

Figura 32. Cilindro doble efecto – dibujo en sección y símbolo
Fuente:(Ebel & Idler, 2019)

2.2.1.2 Motores neumáticos

En los motores neumáticos se consigue el movimiento de rotación de un eje mediante aire a presión. Los dos tipos más utilizados son los motores son: (Gonzáles, 2002b).

- Motores de aletas rotativas.
- Motores de pistones axiales.
- Motores de engranajes.
- Motores de turbina

En la siguiente imagen se puede ver cómo está constituido un motor neumático simple y cuál es su símbolo.

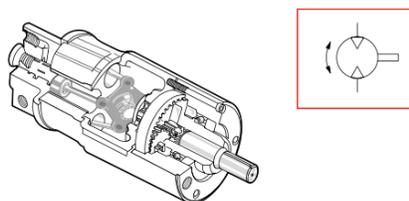


Figura 33. Motor neumático
Fuente:(Ebel & Idler, 2019)

Los motores de pistones axiales tienen un eje de giro solidario a un tambor que se ve obligado a girar ante las fuerzas que ejercen varios cilindros, que se apoyan sobre un plano inclinado. Otro método común más sencillo de obtener movimientos de rotación a partir de actuadores neumáticos se basa en el empleo de cilindros cuyo émbolo se encuentra acoplado a un sistema de piñón-cremallera.

En general y debido a la compresibilidad del aire, los actuadores neumáticos no consiguen una buena precisión de posicionamiento. Sin embargo, su sencillez y robustez hacen adecuado su uso en aquellos casos en los que sea suficiente un posicionamiento en dos situaciones diferentes (todo o nada). Por ejemplo, son utilizados en manipuladores sencillos, en apertura y cierre de pinzas o en determinadas articulaciones de algún robot (como el movimiento vertical del tercer grado de libertad de algunos robots tipo SCARA).

2.2.2 Actuadores hidráulicos

Este tipo de actuadores no se diferencia mucho de los neumáticos. En ellos, en vez de aire se utilizan aceites minerales a una presión comprendida normalmente entre los 50 y 100 bar, llegándose en ocasiones a superar los 300 bar. Existen, como en el caso de los neumáticos, actuadores del tipo cilindro y del tipo motores de aletas y pistones (González, 2002b).

Sin embargo, las características del fluido utilizado en los actuadores hidráulicos marcan ciertas diferencias con los neumáticos. En primer lugar, el grado de compresibilidad de los aceites usados es considerablemente menor al del aire, por lo que la precisión obtenida en este caso es mayor. Por motivos similares, es más fácil en ellos realizar un control continuo, pudiendo posicionar su eje en todo un intervalo de valores (haciendo uso del servocontrol) con notable precisión. Además, las elevadas presiones de trabajo, diez veces superiores a las de los actuadores neumáticos, permiten desarrollar elevadas fuerzas y pares (González, 2002b).

Por otra parte, este tipo de actuadores presenta estabilidad frente a cargas estáticas. Esto indica que el actuador es capaz de soportar cargas, como el peso o una presión ejercida sobre una superficie, sin aporte de energía (para mover el émbolo de un cilindro sería preciso vaciar este de aceite).

También es destacable su elevada capacidad de carga y relación potencia-peso, así como sus características de auto lubricación y robustez (González, 2002b).

Frente a estas ventajas existen ciertos inconvenientes. Por ejemplo, las elevadas presiones a las que se trabaja propician la existencia de fugas de aceite a lo largo de la instalación. Asimismo, esta instalación es más complicada que la necesaria para los actuadores neumáticos y mucho más que para los eléctricos, necesitando de equipos de filtrado de partículas, eliminación de aire, sistemas de refrigeración y unidades de control de distribución.



Figura 34. Motor hidráulico

Fuente: (Xu et al., 2020)

Los accionamientos hidráulicos se usan con frecuencia en aquellos robots que deben manejar grandes cargas (de 70 a 205kg).

2.2.3 Actuadores eléctricos

Las características de control, sencillez y precisión de los accionamientos eléctricos han hecho que sean los más usados en los robots industriales actuales (González, 2002b).

Dentro de los actuadores eléctricos pueden distinguirse tres tipos diferentes:

- Motores de corriente continua (DC).
- Servomotores
- Motores paso a paso
- Motores de corriente alterna (AC)

2.2.3.1 Motores de corriente continua (DC).

Los motores de DC están constituidos por dos devanados internos, inductor e inducido, que se alimentan con corriente continua: (González, 2002b).

El inducido, también denominado devanado de excitación, está situado en el estator y crea un campo magnético de dirección fija, denominado excitación.

El inductor, situado en el rotor, hace girar al mismo debido a la fuerza de Lorentz que aparece como combinación de la corriente circulante por él y del campo magnético de excitación. Recibe la corriente del exterior a través del colector de delgas, en el que se apoyan unas escobillas de grafito. En la imagen siguiente se muestra como aeta conformado un motor de corriente directa

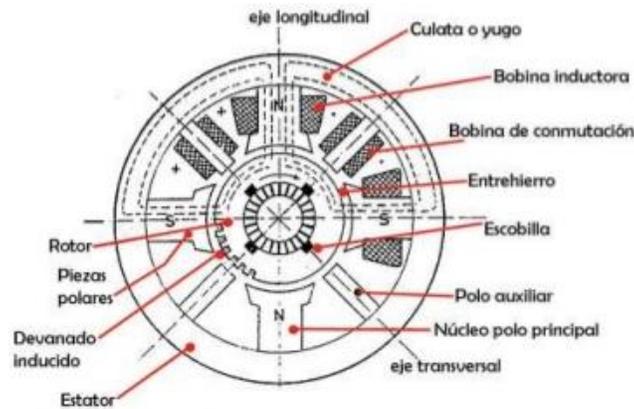


Figura 35. Partes de un Motor e corriente continúa

Fuente: (Mitch Rc, 2014)

En la actualidad los motores de corriente continua son utilizados en proyectos de robótica para procesos móviles de 4 ruedas en la que se necesite velocidad sin tanto torque. En la figura siguiente se puede ver uno de los motores en dc utilizados hoy en día con el controlador arruino.



Figura 36. Partes de un Motor e corriente continúa

Fuente: (PatagoniaTec, s. f.)

Al utilizar este tipo de motores también se necesitará tener un control sobre el ya sea controlado la velocidad o su sentido de giro , es por eso que se utilizan diferentes tipos de drivers de placa eléctrica para poder controlarlos, un ejemplo es la del driver L298N que funciona para el controlador Arduino.

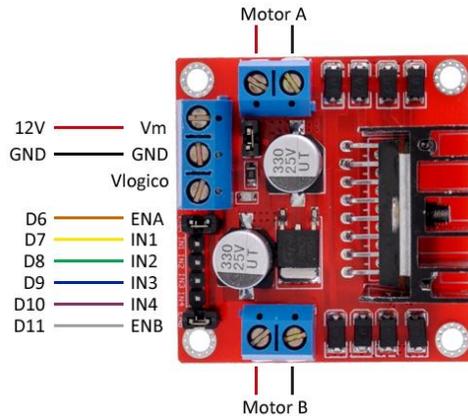


Figura 37. Drive L298N para control de motor e corriente continúa

Fuente: (Llamas, 2016)

Para poder controlar los motores de corriente directa a partir de una placa Arduino se necesitará entonces, una placa de Arduino , un driver de control de motores en dc como se muestra en la siguiente imagen.

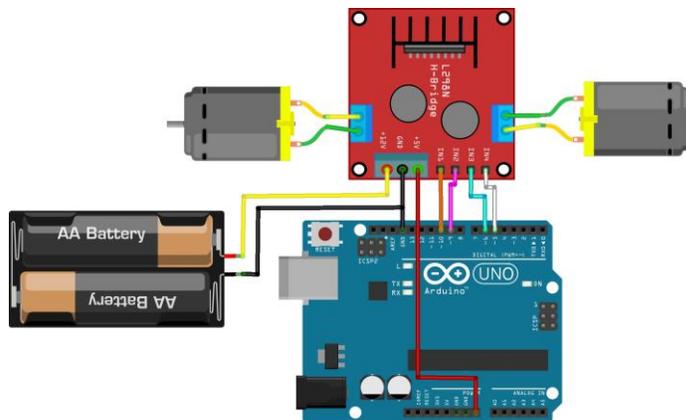


Figura 38. Esquema de conexión para control de motores en dc mediante arduino.

Fuente: (MakerSpace, s. f.)

2.2.3.2 Servomotor

Son los más usados en la actualidad debido a su facilidad de control. En este caso, se utiliza en el propio motor un sensor de posición (Encoder) para poder realizar su control.

Los servomotores funcionan por medio de modulación de ancho de pulso (PWM) Todos los servos disponen de tres cables, dos para alimentación Vcc y Gnd (4.8 a 6 [V]) y un tercero para aplicar el tren de pulsos de control, que hace que el circuito de control diferencial interno ponga el servo en la posición indicada.

La frecuencia usada para mandar la secuencia de pulsos al servomotor es de 50 Hz esto significa que cada ciclo dura 20 ms, Las duraciones de cada pulso se interpretan como comandos de posicionamiento del motor, mientras que los espacios entre cada pulso son despreciados.

Para entender esto, veamos que físicamente el servomotor se compone de tres cables: GND, Power (5v) y Control. Es decir que los dos primeros se usan como alimentación en cuanto el cable de control es usado para enviar la señal de posicionamiento del motor con una secuencia de pulsos PWM.

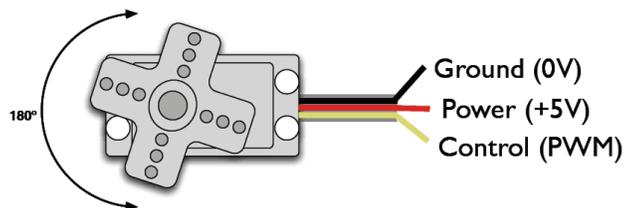


Figura 39. Esquema de conexión para un servomotor .

Fuente: (Castaño, 2019)

Un servo con Arduino podrá tener una potencialidad tremenda, dado que la propia placa de Arduino será la encargada de enviar esa señal PWM para posicionar el motor en una posición en grados en específico.

La señal PWM que ingresa al Servo a través de Arduino, es procesada y dependiendo del ancho de dicho pulso, el servomotor se desplazará a un determinado ángulo. Generalmente los

servomotores comerciales procesan un pulso con un periodo de 20ms para conseguir efectuar todo el movimiento del servo adecuadamente.

En el mercado es posible encontrar varios tipos de servos, y es importante entender sus características para adaptarlos adecuadamente a nuestros proyectos, por lo tanto, es de extrema importancia de que leas la hoja de datos del servo que deseas comprar para ver si se adapta en fuerza y grados de movimiento a tu proyecto.

Existen 2 servos básicos que son:

1. Servomotor de -90 grados a 90 grados (total de 180 grados)
2. Servomotor de 360 grados

Con el servomotor de 180 grados con Arduino podremos controlar la posición donde deseamos ubicar el motor, por otro lado cuando conectamos un servomotor 360 grados en Arduino ya NO tendremos control de posición, porque el servo gira constantemente. Este servomotor de 360 grados es utilizado para realizar control de velocidad, bastante empleado en pequeños o medianos robots para el control de velocidad de las ruedas.

Un factor a tener en cuenta es observar la fuerza del servo, que sea adecuada para mover la carga que necesitamos en nuestro proyecto.

Servomotor de -90 grados a 90 grados (total de 180 grados):

Tomemos como ejemplo un servomotor 180 grados con Arduino. En este caso, para posicionar el motor en 0 grados, nuestro pulso debe ser de 1ms encendido y los 19ms restantes debe estar apagado. Para 90 grados, nuestro pulso debe ser de 1.5ms encendido y los 18.5ms restantes debe estar apagado. Y por último, para 180 grados, nuestro pulso debe ser de 2ms encendido y los 18ms restantes debe estar apagado (Castaño, 2019b).

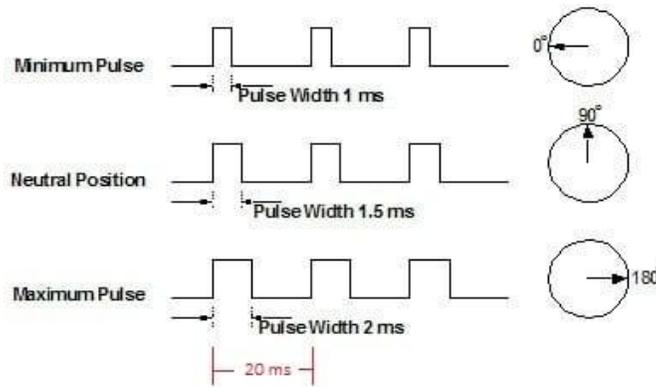


Figura 40. Tiempos de pulso para un servomotor de 180 grados .
Fuente: (Castaño, 2019)

Servomotor de 360 grados:

Para eso vamos a valernos de la misma librería del Arduino solo que en este caso usaremos las siguientes configuraciones para manipular un servo continuo o de 360 grados.

Primero veamos que las señales PWM que deben ser enviadas son exactamente las mismas, con la diferencia de que ya no tendremos el control de posición (CCW: sentido antihorario; CW: Centido Horario (Castaño, 2019).

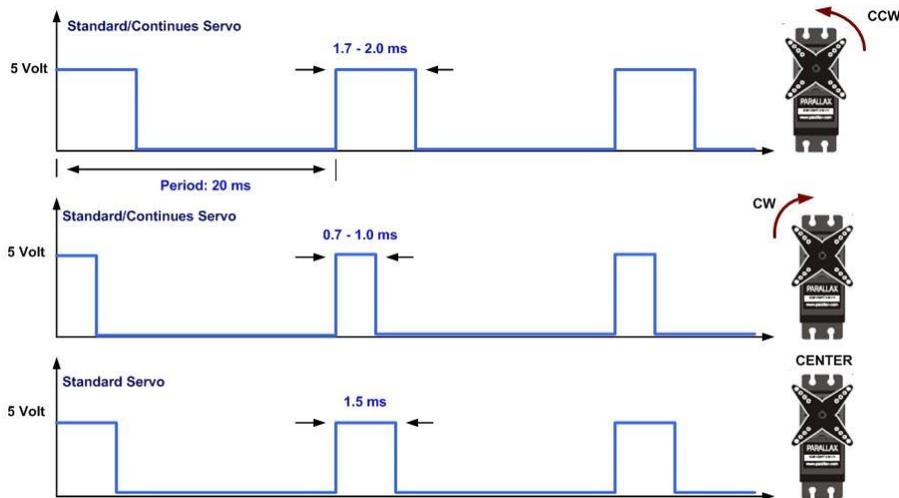


Figura 41. Tiempos de pulso para un servomotor de 360 grados .
Fuente: (Castaño, 2019b)

2.2.3.3 Motor Paso a paso

Los motores paso a paso generalmente no han sido considerados dentro de los accionamientos industriales, debido principalmente a que los pares para los que estaban disponibles eran muy pequeños y los pasos entre posiciones consecutivas eran grandes. En los últimos años se han mejorado notablemente sus características técnicas, especialmente en lo relativo a su control, lo que ha permitido fabricar motores paso a paso capaces de desarrollar pares suficientes en pequeños pasos para su uso como accionamientos industriales (González, 2002b).

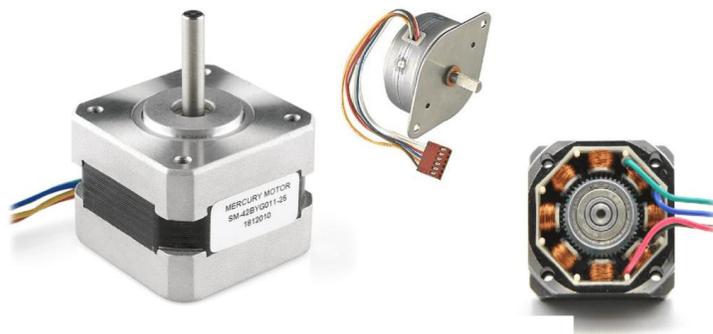


Figura 42. Motores paso a paso.

Fuente: (Castaño, 2019)

Existen tres tipos de motores paso a paso:

- de imanes permanentes
- de reluctancia variable
- híbridos.

En los primeros, de imanes permanentes, el rotor, que posee una polarización magnética constante, gira para orientar sus polos de acuerdo al campo magnético creado por las fases del estator (González, 2002b).

En los motores de reluctancia variable, el rotor está formado por un material ferro-magnético que tiende a orientarse de modo que facilite el camino de las líneas de fuerza del campo magnético generado por las bobinas de estator. No contiene, por tanto, imanes permanentes. El estator es similar a un motor DC de escobillas (González, 2002b).

La reluctancia de un circuito magnético es el equivalente magnético a la resistencia de un circuito eléctrico. La reluctancia del circuito disminuye cuando el rotor se alinea con el polo del estator. Cuando el rotor está en línea con el estator el hueco entre el rotor y el estator es muy pequeño. En este momento la reluctancia está al mínimo (González, 2002b).

La inductancia del bobinado también varía cuando el rotor gira. Cuando el rotor está fuera de alineación, la inductancia es muy baja, y la corriente aumentará rápidamente. Cuando el rotor se alinea con el estator, la inductancia será muy grande. Esta es una de las dificultades del manejo de un motor de reluctancia variable (González, 2002b).

Los motores híbridos combinan el modo de funcionamiento de los dos anteriores.

Bobinas del motor paso a paso

A continuación podemos ver un esquema que muestra la configuración de las diferentes bobinas dentro de un motor paso a paso unipolar y bipolar.

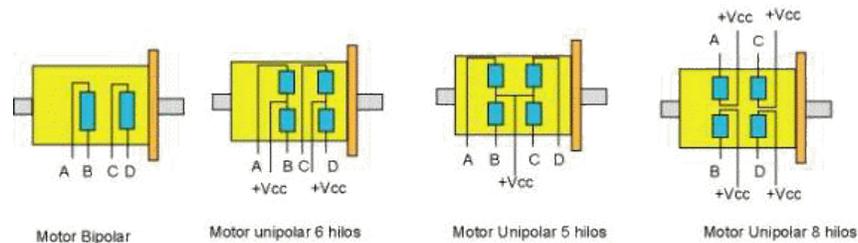


Figura 43. Configuración de bobinas para motores paso a paso.
Fuente: (Castaño, 2019)

Estos motores de paso pueden ser encontrados en diferentes dispositivos comunes tales como las impresoras y muchas veces puede resultar especialmente útil poder reconocer cada uno de los terminales del motor, por lo tanto preparé el siguiente video explicándote como reconocer fácilmente las bobinas y los cables de un motor paso a paso para controlarlo posteriormente con un Arduino.

Motor Paso a Paso Unipolar con Arduino

Este es el motor de paso más común, el cual posee dos bobinas en cada uno de los estatores donde cada una de esas bobinas posee un punto común, por lo tanto este motor típicamente posee 5 o 6 cables. Y viendo esta entrada estarás en la capacidad de controlar un motor paso a paso de 6 cables con Arduino y mucho más (Castaño, 2019).

Los motores unipolares, siempre energizan sus fases de la misma forma. La derivación “común”, siempre será negativa o positiva según nuestro criterio. La otra punta de la derivación contendrá la polarización contraria. Los motores unipolares se pueden implementar con un simple circuito de transistor. La desventaja es que hay menos torque disponible porque solo la mitad de las bobinas se pueden energizar a la vez. En la siguiente figura se puede ver un motor unipolar con el drive ULN2003 (Castaño, 2019).

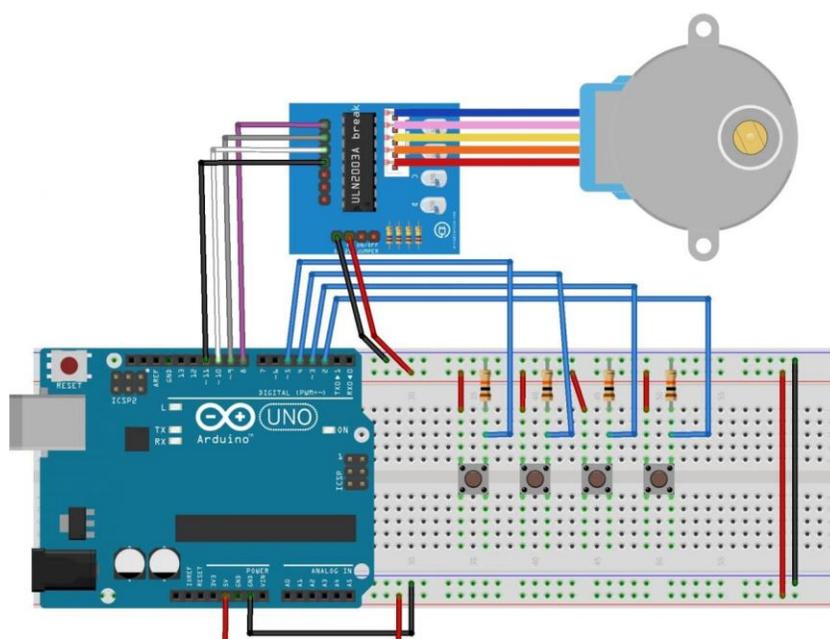


Figura 44. Configuración conexión entre el driver Uln2003 y arduino para el control de un M paso a paso unipolar.
Fuente: (Castaño, 2019)

Motor Paso a Paso Bipolar con Arduino

También posee dos bobinas, con la diferencia que no tienen un punto intermedio común, por lo tanto, es fácil intuir que posee 4 cables. Sin embargo, para conseguir el control de este motor bipolar será necesario efectuar una configuración de 2 puente H usando 8 transistores, o bien usando algún driver o integrado especial para poder mandar la secuencia de pasos (Castaño, 2019).

Los motores bipolares necesitan los circuitos de puente H porque es necesario invertir el flujo de corriente a través de las fases. Al energizar las fases alternando la polaridad, todas las bobinas pueden ponerse a trabajar girando el motor. En la siguiente figura se puede ver el control de un motor paso a paso bipolar con la conexión del driver L298N hacia un Arduino (Castaño, 2019).

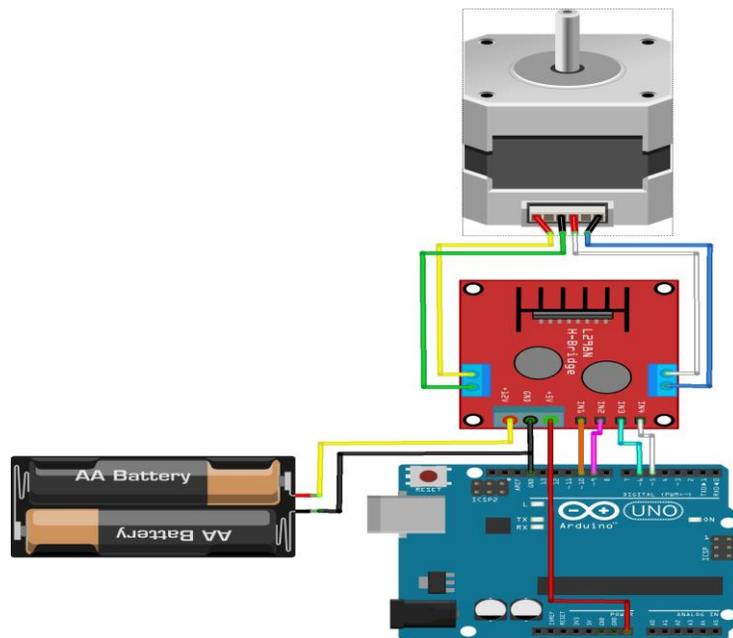


Figura 45. Configuración conexión entre el driver L298n y arduino para el control de un M paso a paso bipolar.
Fuente: (MakerSpace, s. f.)

2.2.3.4 Motores de corriente alterna.

Este tipo de motores no ha tenido aplicación en robótica hasta hace unos años, debido fundamentalmente a la dificultad de su control. Sin embargo, las mejoras que se han introducido en las maquinas síncronas hacen que se presenten como un claro competidor de los motores de corriente continua. Esto se debe principalmente a tres factores: (González, 2002b).

- la construcción de los motores síncronos sin escobillas.
- el uso de convertidores estáticos que permiten variar la frecuencia (y así la velocidad de giro) con facilidad y precisión.
- el empleo de la microelectrónica, que permite una gran capacidad de control.

Existen dos tipos fundamentales de motores de corriente alterna: (González, 2002b).

- motores asíncronos
- motores síncronos

Motores asíncronos de inducción:

Son probablemente los más sencillos y robustos de los motores eléctricos. El rotor está constituido por varias barras conductoras dispuestas paralelamente al eje del motor y por dos anillos conductores en los extremos. El conjunto es similar a una jaula de ardilla y por eso se le denomina también motor de jaula de ardilla. El estator consta de un conjunto de bobinas, de modo que cuando la corriente alterna trifásica las atraviesa, se forma un campo magnético rotatorio en las proximidades del estator. Esto induce corriente en el rotor, que crea su propio campo magnético. La interacción entre ambos campos produce un par en el rotor. No existe conexión eléctrica directa entre estator y rotor (González, 2002b).

Motores síncronos:

El motor síncrono, como su nombre indica, opera exactamente a la misma velocidad que el campo del estator, sin deslizamiento.

El inducido se sitúa en el rotor, que tiene polaridad constante (imanes permanentes o bobinas), mientras que el inductor situado en el estator, está formado por tres devanados iguales decalados 120° eléctricos y se alimenta con un sistema trifásico de tensiones. Es preciso resaltar la similitud existente entre este esquema de funcionamiento y el del motor sin escobillas (González, 2002b).

En los motores síncronos la velocidad de giro depende únicamente de la frecuencia de la tensión que alimenta el inducido. Para poder variar esta precisión, el control de velocidad se realiza mediante un convertidor de frecuencia. Para evitar el riesgo de pérdida de sincronismo se utiliza un sensor de posición continuo que detecta la posición del rotor y permite mantener en todo momento el ángulo que forman los campos del estator y rotor. Este método de control se conoce como autosíncrono o autopilotado (González, 2002b).

2.2.4 Comparación entre los diferentes tipos de actuadores

Como resumen de los actuadores utilizados en robótica se presenta la siguiente tabla:

Tabla 12

Tabla de resumen de actuadores eléctricos

| Características de los distintos tipos de actuadores para robots | | | |
|---|--|--|---|
| | Neumáticos | Hidráulicos | Eléctricos |
| Energía | Aire a presión (5-10 bar) | Aceite mineral (50-100 bar) | Corriente eléctrica |
| Opciones | Cilindros Motor de paletas Motor de pistón | Cilindros Motor de paletas Motor de pistones axiales | Corriente continua Corriente alterna Motor paso a paso Servomotor |
| Ventajas | Baratos Rápidos Sencillos Robustos | Rápidos Alta relación potencia- peso Autolubricantes Alta capacidad de carga Estabilidad frente a cargas estáticas | Precisos Fiables Fácil control Sencilla instalación Silenciosos |
| Desventajas | Dificultad de control continuo Instalación especial (compresor, filtros) Ruidoso | Difícil mantenimiento Instalación especial (filtros, eliminación aire) Frecuentes fugas Caros | Potencia limitada |

Fuente: (González, 2002b)

2.3 Sensores internos (Posición, velocidad y presencia)

Para conseguir que un robot realice su tarea con la adecuada precisión, velocidad e inteligencia, será preciso que tenga conocimiento tanto de su propio estado como del estado de su entorno. La información relacionada con su estado (fundamentalmente la posición de

sus articulaciones) la consigue con los denominados sensores internos, mientras que la que se refiere al estado de su entorno, se adquiere con los sensores externos (González, 2002b). Para clasificar los sensores de acuerdo a los tipos existentes podemos ver la siguiente tabla:

Tabla 13*Tabla de tipos de sensores internos para robots*

| Tipos de sensores internos de robots | | |
|--------------------------------------|------------|--|
| Posición | Analógicos | Potenciómetros <i>Resolver</i> Sincro <i>Inductosyn</i> LVDT |
| | Digitales | <i>Encoders</i> absolutos <i>Encoders</i> incrementales Regla óptica |
| Velocidad | | Taco-generatriz |
| Presencia | | Inductivo Capacitivo Efecto Hhall Célula Reed Óptico Ultrasonidos Contacto |

Fuente: (González, 2002b)

2.3.1 Sensores de posición

Para el control de posición angular se emplean fundamentalmente los denominados encoders y resolvers. Los potenciómetros dan bajas prestaciones por lo que no se emplean salvo en contadas ocasiones (robots educacionales, ejes de poca importancia) (González, 2002b).

2.3.1.1 Codificadores angulares de posición (encoders)

Los codificadores ópticos o encoders incrementales constan, en su forma más simple, de un disco transparente con una serie de marcas opacas colocadas radialmente y equidistantes entre sí, de un sistema de iluminación en el que la luz es colimada de forma adecuada, y de un elemento fotorreceptor. El eje cuya posición se quiere medir va acoplado al disco transparente. Con esta disposición, a medida que el eje gire se irán generando pulsos en el receptor cada vez que la luz atraviese cada marca, y llevando una cuenta de estos pulsos es posible conocer la posición del eje (González, 2002b).

Existe, sin embargo, el problema del desconocimiento en un momento dado de si se está realizando un giro en un sentido o en el opuesto, con el peligro que supone no estar contando adecuadamente. Una solución a este problema consiste en disponer de otra franja de marcas, desplazada de la anterior de manera que el tren de pulsos que con ella se genere esté desplazado 90° eléctricos con respecto al generado por la primera franja. De esta manera, con un circuito relativamente sencillo, es posible obtener una señal adicional que indique cuál es el sentido de giro y que actúe sobre el contador correspondiente indicándole que incremente o reduzca la cuenta que se está realizando (González, 2002b).

Es necesario además disponer de una marca de referencia sobre el disco que indique que se ha dado una vuelta completa y que, por tanto, se ha de empezar la cuenta de nuevo. Esta marca sirve también para poder comenzar a contar tras recuperarse de una caída de tensión (González, 2002b).

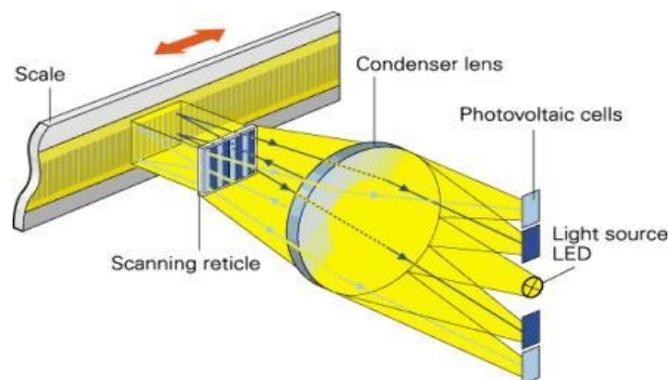


Figura 46. Esquema de funcionamiento del codificador angular de posición encoder
Fuente: (González, 2002)

La resolución de este tipo de sensores depende directamente del número de marcas que se pueden poner físicamente en el disco. Un método relativamente sencillo para aumentar esta resolución es, no solamente contabilizar los flancos de subida de los trenes de pulsos, sino contabilizar también los de bajada, incrementando así la resolución del captador, pudiéndose llegar, con ayuda de circuitos adicionales, hasta 100.000 pulsos por vuelta (González, 2002b).

El funcionamiento básico de los codificadores o encoders absolutos es similar al de los incrementales. Se tiene una fuente de luz con las lentes de adaptación correspondientes, un disco graduado y unos fotoreceptores. En este caso, el disco transparente se divide en un número determinado de sectores (potencia de 2), codificándose cada uno de ellos según un código binario cíclico (normalmente código de Gray) que queda representado por zonas transparentes y opacas dispuestas radialmente. No es necesario ahora ningún contador o electrónica adicional para detectar el sentido del giro, pues cada posición (sector) es codificado de forma absoluta. Su resolución es fija, y vendrá dada por el número de anillos que posea el disco graduado. Las resoluciones habituales van desde 28 a 219 bits (desde 256 a 524288 posiciones distintas) (González, 2002b).

Normalmente los sensores de posición se acoplan al eje del motor. Considerando que en la mayor parte de los casos entre el eje del motor y el de la articulación se sitúa un reductor de relación N , cada movimiento de la articulación se verá multiplicado por N al ser medido por el sensor. Éste aumentara así su resolución, multiplicándola por N (González, 2002b).

Los encoders pueden presentar problemas mecánicos debido a la gran precisión que se debe tener en su fabricación. La contaminación ambiental puede ser una fuente de interferencias en la transmisión óptica. Son dispositivos particularmente sensibles a golpes y vibraciones, estando su margen de temperatura de trabajo limitado por la presencia de componentes electrónicos (González, 2002b).

En arduino se tiene el sensor FC-03 para poder sensar pulsos de velocidad angular, una posible conexión en Arduino puede ser :

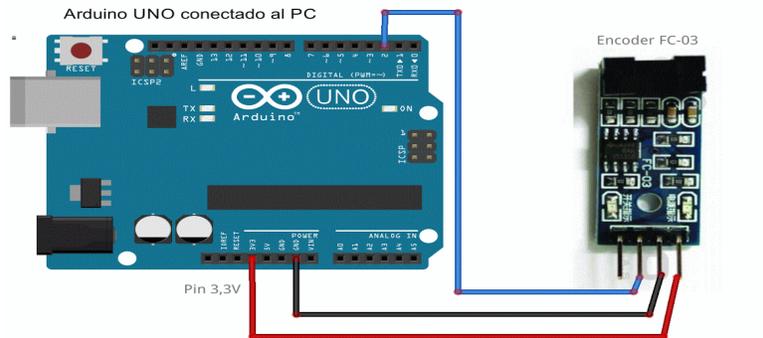


Figura 47. Esquema de conexión de un encoder en arduino
Fuente: (Dutour, 2019)

2.3.1.2 Captadores angulares de posición (sincro-resolvers).

La otra alternativa en sensores de posición para robots la representan los resolvers y los sincroresolvers, también llamados sincros. Se trata de sensores analógicos con resolución teóricamente infinita. El funcionamiento de los resolvers se basa en la utilización de una bobina solidaria al eje excitada por una portadora, generalmente con 400Hz, y por dos bobinas fijas situadas a su alrededor (González, 2002b).

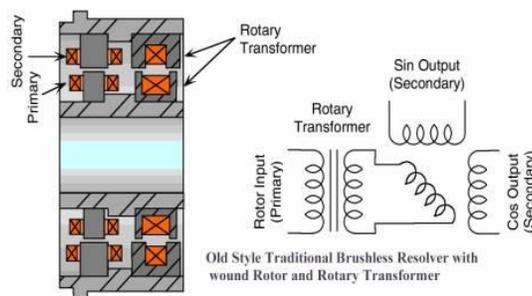


Figura 48. Esquema de un sincro-resolvers , estilo antiguo tradicional sin escobillas
Fuente: (González, 2002)

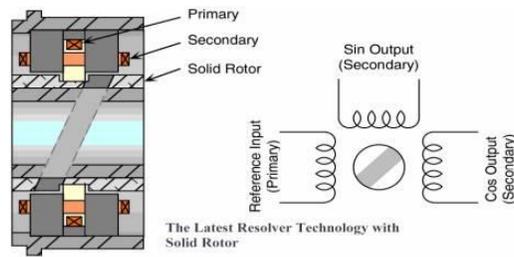


Figura 49. Esquema de un micro-resolvers , estilo actual con tecnología de rotor sólido
Fuente: (González, 2002)

El giro de la bobina móvil hace que el acoplamiento con las bobinas fijas varíe, consiguiendo que la señal resultante en éstas dependa del seno del ángulo de giro. La bobina móvil excitada con tensión $V \sin(\omega t)$ y girada un ángulo θ induce en las bobinas fijas situadas en cuadratura las siguientes tensiones:

$$V_1 = V \sin(\omega t) \sin \theta$$

$$V_2 = V \sin(\omega t) \cos \theta$$

que es llamada representación del ángulo θ en formato sincro.

El cambio del llamado formato sincro a formato resolver o viceversa es inmediato, ya que se puede pasar de uno a otro a través de la llamada red de Scott, o transformador de Scott, o funcionamiento bidireccional. Para poder tratar el sistema de control, la información generada por los resolvers y los sincros es necesario convertir las señales analógicas en digitales. Para ello se utilizan los llamados convertidores resolver/digital (r/d), que tradicionalmente se basan en dos tipos de estructuras distintas: tracking y muestreo (sampling) (González, 2002b).

Comparación de sensores de posición angular:

Tabla 14

Tabla diferentes sensores de posición angular

| Comparación entre diferentes sensores de posición angular | | | | |
|---|-------------------|----------------|------------|---------------------|
| | Robustez mecánica | Rango dinámico | Resolución | Estabilidad térmica |
| Encoder | mala | media | buena | buena |
| Resolver | buena | buena | buena | buena |
| Potenciómetro | regular | mala | mala | mala |

Fuente: (González, 2002)

2.3.1.3 Sensores lineales de posición (LVDT)

Entre los sensores de posición lineales destaca el transformador diferencial de variación lineal (LVDT) debido a su casi infinita resolución, poco rozamiento y alta repetibilidad. Su funcionamiento se basa en la utilización de un núcleo de material ferromagnético unido al eje cuyo movimiento se quiere medir. Este núcleo se mueve linealmente entre un devanado primario y dos secundarios, haciendo con su movimiento que varíe la inductancia entre ellos (González, 2002b).

Los dos devanados secundarios conectados en oposición serie ven como la inducción de la tensión alterna del primario, al variar la posición del núcleo, hace crecer la tensión de un devanado y disminuir la del otro. Del estudio de la tensión se deduce que ésta es proporcional a la diferencia de inductancias mutuas entre el devanado primario con cada uno de los secundarios, y que por tanto depende linealmente del desplazamiento del vástago solidario al núcleo (González, 2002b).

Además de las ventajas señaladas, el LVDT presenta una alta linealidad, gran sensibilidad y una respuesta dinámica elevada. Su uso está ampliamente extendido, a pesar del inconveniente de poder ser aplicado únicamente en la medición de pequeños desplazamientos.

Otros sensores lineales que también se emplean con relativa frecuencia son las denominadas reglas ópticas (equivalentes a los codificadores ópticos angulares) y las reglas magnéticas o Inductosyn. El funcionamiento del Inductosyn es similar a la del resolver con la diferencia de que el rotor desliza linealmente sobre el estator (González, 2002b).

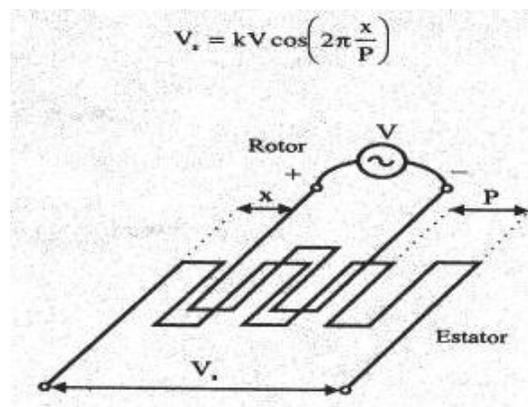


Figura 50. Esquema de funcionamiento de un Inductosyn
Fuente: (González, 2002)

El estator se encuentra excitado por una tensión conocida que induce en el rotor dependiendo de su posición relativa una tensión V_s .

2.3.2 Sensores de velocidad

La captación de la velocidad se hace necesaria para mejorar el comportamiento dinámico de los actuadores del robot. La información de la velocidad de movimiento de cada actuador se realimenta normalmente a un bucle de control analógico implementado en el propio accionador del elemento motor. No obstante, en las ocasiones en las que el sistema de control del robot lo exija, la velocidad de giro de cada actuador es llevada hasta la unidad de control del robot (González, 2002b).

Normalmente, y puesto que el bucle de control de velocidad es analógico, el sensor usado es un taco generatriz que proporciona una tensión proporcional a la velocidad de giro de su eje (10 mV por rpm) (González, 2002b).

Otra posibilidad, usada para el caso de que la unidad de control del robot precise conocer la velocidad de giro de las articulaciones, consiste en derivar la información de posición que ésta posee.

2.3.3 Sensores de presencia

Este tipo de sensor es capaz de detectar la presencia de un objeto dentro de un radio de acción determinado. Esta detección puede hacerse con o sin contacto con el objeto. En el segundo caso se utilizan diferentes principios físicos para detectar la presencia, dando lugar a los diferentes tipos de sensores (González, 2002b).

En el caso de detección con contacto, se trata siempre de un interruptor, normalmente abierto o normalmente cerrado según interese, actuando mecánicamente a través de un vástago u otro dispositivo. Los detectores de presencia se utilizan en robótica principalmente como auxiliares de los detectores de posición, para indicar los límites de las articulaciones y permitir localizar la posición de referencia de cero de éstos en el caso de que sean incrementales (González, 2002b).

Además de esta aplicación, los sensores de presencia se usan como sensores externos, siendo muy sencillos de incorporar al robot por su carácter binario y su costo reducido. Los detectores inductivos permiten detectar la presencia o contar el número de objetos metálicos sin necesidad de contacto. Presentan el inconveniente de distinto comportamiento según del tipo de metal del que se trate. El mismo tipo de aplicación tienen los detectores capacitivos, más voluminosos, aunque en este caso los objetos a detectar no precisan ser metálicos. En cambio presentan problemas de trabajo en condiciones húmedas y con puestas a tierra defectuosa (González, 2002b).

En electrónica con Arduino se puede utilizar el mismo sensor encoder ya sea el Fc-03 o el Lm393, todo dependerá como se programe y la relación que se le de los datos en la programación.

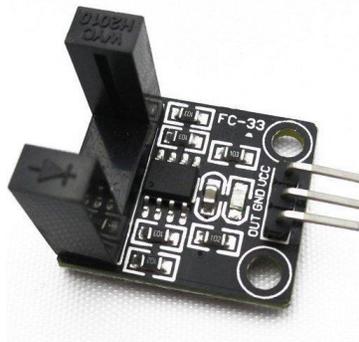


Figura 51. Sensor Lm393 de velocidad para arduino
Fuente: (PatagoniaTec, s. f.)

Los sensores basados en el efecto Hall detectan la presencia de objetos ferromagnéticos por la deformación que estos provocan sobre un campo magnético (González, 2002b).

Los sensores ópticos, sin embargo, pueden detectar la reflexión del rayo de luz procedente del emisor sobre el objeto.

Los sensores/conmutadores Reed de proximidad (frecuentemente referidos como sensores magnéticos) son muy tolerantes al desalineamiento y se ajustan bien a entornos contaminados por polvo y líquido. Constan de dos partes, el conmutador reed y el actuador magnético. El conmutador reed cambia su estado cuando el actuador magnético se acerca a él, sin necesidad de que exista contacto físico entre ambos. La distancia de operación puede variarse con una adecuada elección

del actuador magnético. Las configuraciones del conmutador con contactos normalmente abiertos o intercambiables. En la siguiente figura se muestra como trabaja el sensor magnético reed.

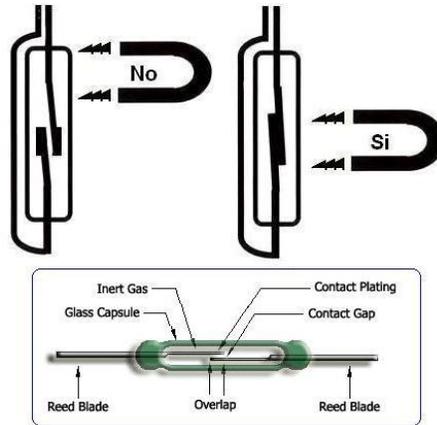


Figura 52. Esquema de un sensor reed magnético
Fuente: (shoptronica, s. f.)

Este tipo de sensores también se los usa en la electrónica con arduinos, se lo designa como LM393 reed:

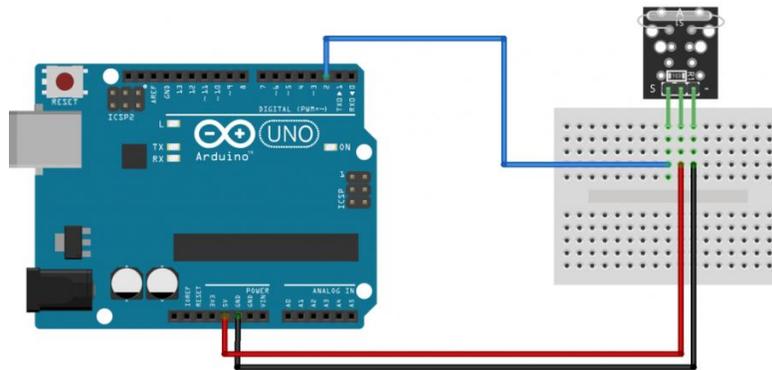


Figura 53. Esquema de conexión con Arduino sensor reed magnético
Fuente: (ArduinoModules, 2021)

2.4 Elementos terminales

Para las aplicaciones industriales, las capacidades del robot básico deben aumentarse por medio de dispositivos adicionales. Podríamos denominar a estos dispositivos como los periféricos del robot. En robótica, el termino de actuador final se utiliza para describir la mano o herramienta que está unida a la muñeca. El actuador final representa la herramienta especial que permite al robot

de uso general realizar una aplicación particular, y debe diseñarse específicamente para dicha aplicación (González, 2002b).

Los actuadores finales pueden dividirse en dos categorías:

- pinzas (gripper)
- herramientas

Las pinzas se utilizan para tomar un objeto, normalmente la pieza de trabajo, y sujetarlo durante el ciclo de trabajo del robot. Hay una diversidad de métodos de sujeción que pueden utilizarse, además de los métodos mecánicos obvios de agarre de la pieza entre dos o más dedos. Estos métodos suplementarios incluyen el empleo de casquetes de sujeción, imanes, ganchos, y cucharas (González, 2002b).

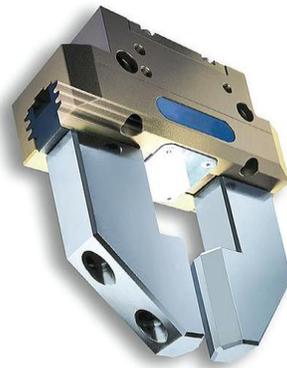


Figura 54. Griper eléctrico
Fuente: (mobile, 2013)

Una herramienta se utiliza como actuador final en aplicaciones en donde se exija al robot realizar alguna operación sobre la pieza de trabajo. Estas aplicaciones incluyen la soldadura por puntos, la soldadura por arco, la pintura por pulverización y las operaciones de taladro. En cada caso, la herramienta particular está unida a la muñeca del robot para realizar la operación (González, 2002b).



Figura 55. Herramienta terminal para suelda
Fuente: (González, 2002)

2.4.1 Pinzas (gripper)

Los elementos de sujeción se utilizan para agarrar y sostener los objetos, y se suelen denominar pinzas. Se distingue entre las que utilizan dispositivos de agarre mecánico y las que utilizan algún otro tipo de dispositivo (ventosas, pinzas magnéticas, adhesivas, ganchos, etc.). Se pueden clasificar según el sistema de sujeción empleado (González, 2002b).

Tabla 15

Tabla de sistemas de sujeción para robots

| Sistemas de sujeción para robots | | |
|---|-----------------------|--|
| Tipo | Accionamiento | Uso |
| Pinza de presión - des. angular - des. lineal | Neumático o eléctrico | Transporte y manipulación de piezas sobre las que no importé presionar |
| Pinza de enganche | Neumático o eléctrico | Piezas grandes dimensiones o sobre las que no se puede ejercer presión |
| Ventosa de vacío | Neumático | Cuerpos con superficie lisa poco porosa (cristal, plástico etc.) |
| Electroimán | Eléctrico | Piezas ferromagnéticas |

Fuente: (González, 2002)

El accionamiento neumático es el más utilizado por ofrecer mayores ventajas en simplicidad, precio y fiabilidad, aunque presenta dificultades de control de posiciones intermedias. En ocasiones se utilizan accionamientos de tipo eléctrico.

En la pinza se suelen situar sensores para detectar el estado de la misma (abierto o cerrado). Se pueden incorporar a la pinza otro tipo de sensores para controlar el estado de la pieza, sistemas de visión que incorporen datos geométricos de los objetos, detectores de proximidad, sensores fuerza par, etc (González, 2002b).

Existen ciertos elementos comerciales que sirven de base para la pinza, siendo posible a partir de ellos diseñar actuadores válidos para cada aplicación concreta. Sin embargo, en otras ocasiones el actuador debe ser desarrollado íntegramente, constituyendo un porcentaje importante dentro del coste total de la aplicación (González, 2002b).

Los tipos de pinzas más comunes pertenecen al **tipo llamado pivotante**. Los dedos de la pinza giran en relación con los puntos fijos del pivote. De esta manera, la pinza se abre y se cierra.

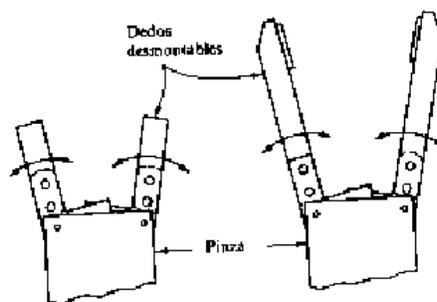


Figura 56. Pinzas pivotantes
Fuente: (González, 2002)

Otro tipo de pinzas se denominan de **movimiento lineal**. En este caso, los dedos se abren y se cierran ejecutando un movimiento paralelo entre sí.

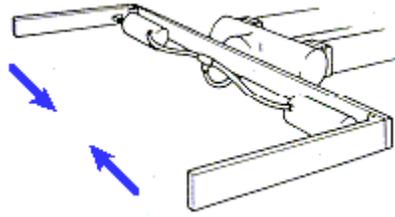


Figura 57. Pinzas pivotantes
Fuente: (González, 2002)

En la elección o diseño de una pinza se han de tener en cuenta diversos factores. Entre los que afectan al tipo de objeto y de manipulación a realizar destacan el peso, la forma, el tamaño del objeto y la fuerza que es necesario ejercer y mantener para sujetarlo. Entre los parámetros de la pinza cabe destacar su peso (que afecta a las inercias del robot), el equipo de accionamiento y la capacidad de control (González, 2002b).

Una regla general es que la pinza debe sujetar a la pieza de trabajo por su centro de gravedad; esto ocasiona que se anulen los momentos que se pudieran generar por el peso de la pieza de trabajo (González, 2002b).

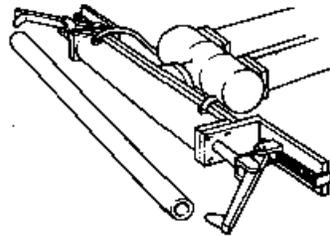


Figura 58. Pinzas pivotantes, en sujeción por centro de gravedad del objeto
Fuente: (González, 2002)

Para reducir los tiempos de ciclo en operaciones de carga y descarga de piezas a máquinas-herramientas se pueden diseñar actuadores finales con doble pinza (González, 2002b).

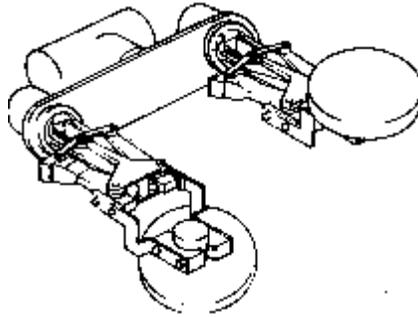


Figura 59. Actuador final con doble pinza.
Fuente: (González, 2002)

Existen otros tipos de pinzas como ventosas, pinzas magnéticas y pinzas adhesivas

Two Finger Gripper - as the pneumatic cylinder is actuated, the fingers move together and apart.

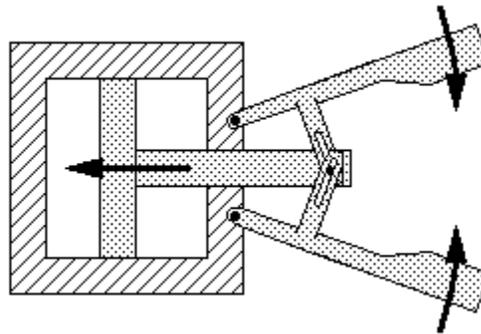


Figura 60. Dos dedos pinza neumáticos.
Fuente: (González, 2002)

Parallel finger actuator - as the cylinder is actuated, the fingers move together and apart in parallel.

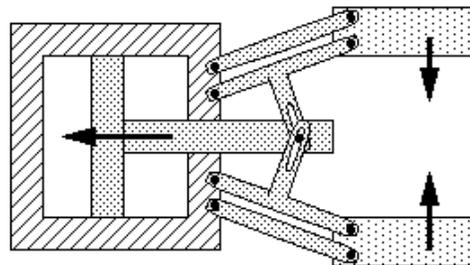


Figura 61. Actuador de dedos paralelos neumáticos.
Fuente: (González, 2002)

Two Fingereed Pneumatic Actuated - as the cylinder is actuated, it translates to the fingers opening or closing. The extra links help increase holding force.

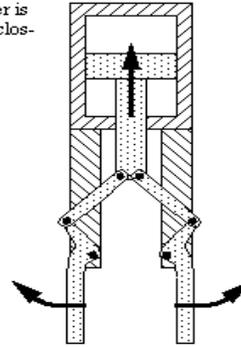


Figura 62. Actuador neumático de dos dedos.

Fuente: (González, 2002)

Two Finger Internal Gripper - as the cylinder is actuated, the fingers move outward.

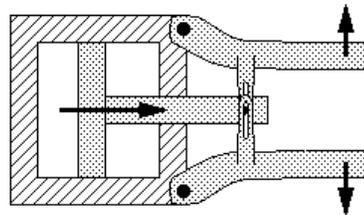


Figura 63. Pinza interna de dos dedos.

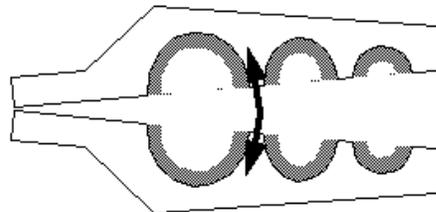
Fuente: (González, 2002)



Self Aligning Finger Pads - small rocking pads are placed on the end fingers, these are also covered with a high friction material, such as rubber. These allow some locational inaccuracy when grasping parts.

Figura 64. Pinza de almohadillas de dedo autoalineado.

Fuente: (González, 2002)



Multiple Part Gripper - the gripper has a number of holes cut for different parts. In this case the gripper can hold three different radii, and the rubber lining will help hold the part.

Figura 65. Pinza de piezas multiple.

Fuente: (González, 2002)

2.4.2 Herramientas

En muchas ocasiones el robot ha de realizar operaciones que no consisten en manipular objetos, sino que implica el uso de una herramienta. Aparte de estos elementos de sujeción y herramientas más o menos convencionales, existen interesantes desarrollos e investigaciones, muchos de ellos orientados a la manipulación de objetos complicados y delicados. Por ejemplo pinzas dotadas de tacto (González, 2002b).

Tabla 16

Tabla de herramientas terminales para robots

| Herramientas terminales para robots | |
|--|--|
| Tipo | Comentarios |
| Pinza soldadura por puntos | Dos electrodos que se cierran sobre la pieza de soldar |
| Soplete soldadura de arco | Aportan el flujo de electrodo que se funde |
| Cucharón para colada | Para trabajos de fundición |
| Atornillador | Suelen incluir la alimentación de tornillos |
| Fresa-lija | Para perfilar, eliminar rebabas, pulir, etc |
| Pistola de pintura | Para pulverización de la pintura |
| Cañón láser | Para corte de materiales, soldadura o inspección |
| Cañón de agua a presión | Para corte de materiales |

Fuente: (González, 2002)

Algunos de los diferentes tipos de herramientas que utilizan los manipuladores se mencionan en las siguientes imágenes (González, 2002b).



Figura 66. Herramienta para impresión 3d.
Fuente: (González, 2002)



Figura 67. Herramienta para soldadura.
Fuente: (González, 2002)

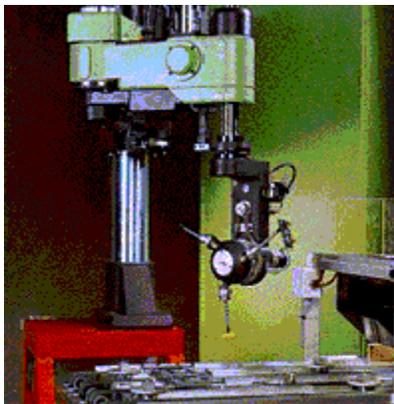


Figura 67. Herramienta para control de calidad.
Fuente: (González, 2002)

ACTIVIDADES UNIDAD 2

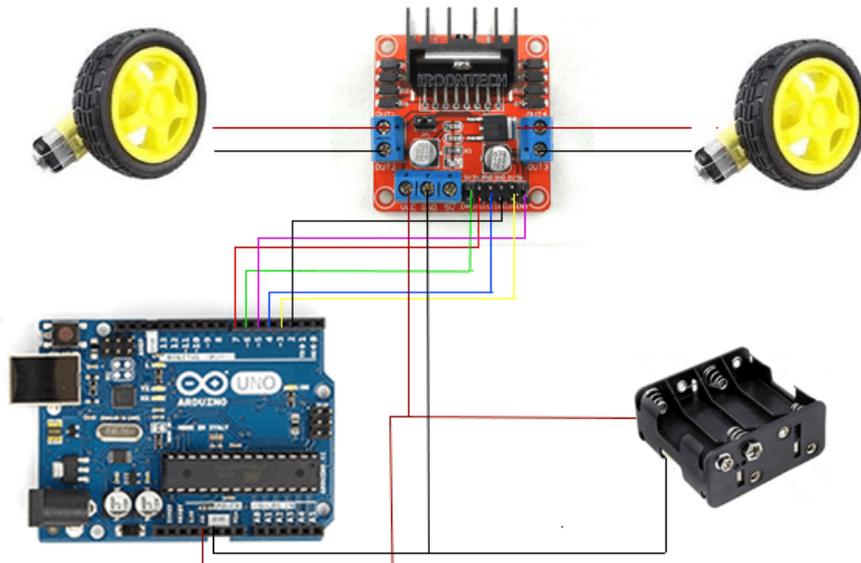
Actividad 1: Preguntas

- 1.- Qué se entiende por transmisor mecánico y cuál son sus características básicas en la robótica?
- 2.- Según su entrada y salida cuáles son los tipos de transmisiones para robots.
- 3.- Describa un mecanismo de conversión de movimiento de circular a lineal.
- 4.- Describa un mecanismo de conversión de movimiento de lineal a circular.
- 5.- Según valores típicos cuál es la relación para los reductores con característica de rigidez torsional?
- 6.- Cómo se puede encontrar el rendimiento a partir del par y velocidad de entrada y salida en un reductor?
- 7.- Cuáles son las características del reductor Harmonic Drive.
- 8.- Según qué características se puede evaluar la selección de un actuador?
- 9.- Cuál es la principal diferencia entre un actuador neumático de cilindro simple efecto con uno de doble efecto.
- 10.- Qué tipos de motores neumáticos considera que se pueden utilizar en procesos donde demanden precisión en el posicionamiento.
- 11.- Cuál es la ventaja de utilizar un servo motor con relación a un motor de corriente continua.
- 12.- Cuál es la desventaja de utilizar un servomotor en relación a un motor paso a paso.
- 13.- Cuál es el drive que me sirve para controlar la velocidad e inversión de giro de un motor utilizando un Arduino.
- 14.- Qué tipo de señal me entrega un servo motor cuando está en funcionamiento.
- 15.- Cuál es el ancho de pulso en ms que debería generar un servomotor para un control de 90 grados, para un servo motor de control de 0 a 180 grados.
- 16.- Cuál es el ancho de pulso en ms que deberá generar un servo motor para un sentido antihorario, para un servomotor continuo de 360 grados
- 17.- Cuál es la configuración de bobinas para un motor paso a paso unipolar de 8 hilos.
- 18.- Qué drive se puede utilizar para controlar desde Arduino un motor paso a paso unipolar de 5 hilos.
- 19.- ¿Se puede utilizar el drive de control de velocidad para motores de corriente directa, para poder ahora controlar un motor paso a paso bipolar de 4 hilos?

- 20.- Qué tipos de sensores se pueden utilizar para medir la posición.
- 21.- Cómo funciona un encoder , y cuál es la numeración del integrado para utilizarlo en arduino?
- 22.- Para que me sirve el sensor Lm393?
- 23.- Cómo funciona un sensor reed magnético?
- 24.- Cuál es la diferencia entre un griper y una herramienta accionamiento terminal?
- 25.- Dibuje un griper por pinza de dos dedos.

Actividad 2: Actividades de simulación/ prácticas.

1.- Se desea dar implementación al siguiente circuito, se pide realizar una programación básica para poner en funcionamiento los dos motores de corriente continua.



Fuente: (Autores)

2.- Realizar un esquema de un circuito para controlar dos servo motores desde Arduino, en la que se desea que mediante un potenciómetro controlar el movimiento del servo A, y que el Servo B funcione al lado contrario del servo A. se pide realizar el esquema y el programa en Arduino.

3.- Realizar el esquema para el control de un motor paso a paso Bipolar, en la que se active cuando un sensor de cualquier tipo le una magnitud física.

Se pide realizar el esquema electrónico y el programa de control.

4.- Se pide acondicionar el circuito de la figura 1, para que el sistema funcione como un carro de juguete, en la que se deberá activar cuando el carro vea una luz blanca y se desactive cuando este a obscuras o sin luz.

UNIDAD 3: PROGRAMACIÓN

3.1 Grados de Libertad GDL

En base a lo visto anteriormente en el apartado 1.3.1, antes de empezar a programar el movimiento de 2 a 3 GDL de un robot, se tendrá que verificar los el número GDL de los siguientes ejemplos.

Grados de libertad en mecanismos en un plano

Ecuación de Gruebler:

$$M = 3L - 2J - 3G$$

M= grados de libertad o movilidad
L=numero de eslabones
J=numero de juntas
G=numero de eslabones fijos.

Ecuación de Grubler modificada Kutzbach.

$$M = 3(L-1) - 2J_1 - J_2$$

M=grados de libertad o movilidad
L=numero de eslabones
J₁=numero de 1 GDL, juntas completas
J₂=numero de 2 GDL, semijuntas.

Figura 68. Grados de Libertad en mecanismos en un plano.
Fuente: (Nuñez, 2020)

Ejemplo 1:

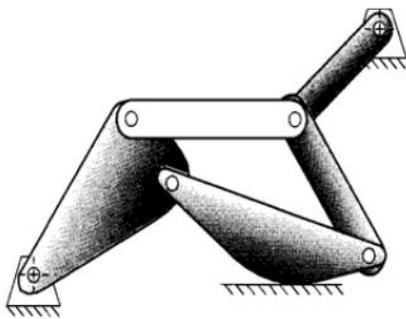


Figura 69. Mecanismo ejemplo 1.
Fuente: (Nuñez, 2020)

Aplicando la ecuación de Grubler modificada

$$M=3(L-1)-2J_1-J_2$$

$$L=6$$

$$J_1=7$$

$$J_2=1$$

$$GDL=0$$

Ejemplo 2:

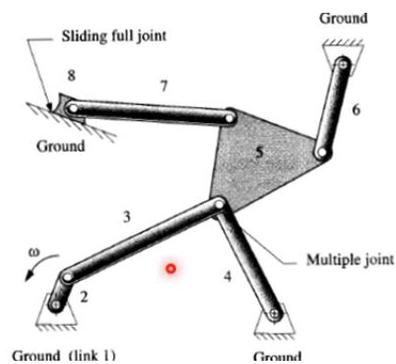


Figura 70. Mecanismo ejemplo 2.
Fuente: (Nuñez, 2020)

Ecuación de Grubler modificada

$$M=3(L-1)-2J_1-J_2$$

$$L=8$$

$$J_1=10$$

$$J_2=0$$

$$GDL=1$$

En base a los ejemplos anteriores, se pide analizar al siguiente esquema dinámico de revolutas de 3 GDL.

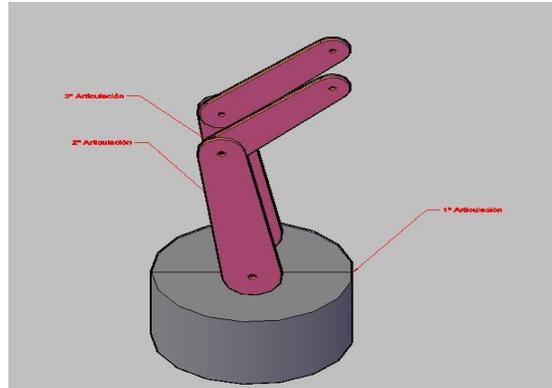


Figura 69. Mecanismo robot 3 GDL.
Fuente: (Nuñez, 2020)

Para poder empezar a programar un brazo de 3 GDL en forma básica utilizando servomecanismos es necesario conocer el apartado de servomotores de la unidad 2. En este tema se profundizará conceptos de programación utilizando el IDE de Arduino.

3.1.1 Servo Arduino: Control de Posición

La señal PWM que ingresa al Servo a través de Arduino, es procesada y dependiendo del ancho de dicho pulso, el servomotor se desplazará a un determinado ángulo.

Generalmente los servomotores comerciales procesan un pulso con un periodo de 20ms para conseguir efectuar todo el movimiento del servo adecuadamente.

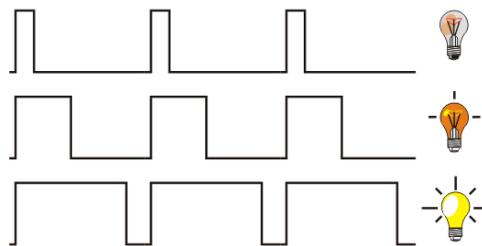


Figura 70. Señal referencial del PWM
Fuente: (Nuñez, 2020)

Tomemos como ejemplo un servomotor 180 grados con Arduino. En este caso, para posicionar el motor en 0 grados, nuestro pulso debe ser de 1ms encendido y los 19ms restantes debe estar apagado. Para 90 grados, nuestro pulso debe ser de 1.5ms encendido y los 18.5ms restantes debe

estar apagado. Y por último, para 180 grados, nuestro pulso debe ser de 2ms encendido y los 18ms restantes debe estar apagado.

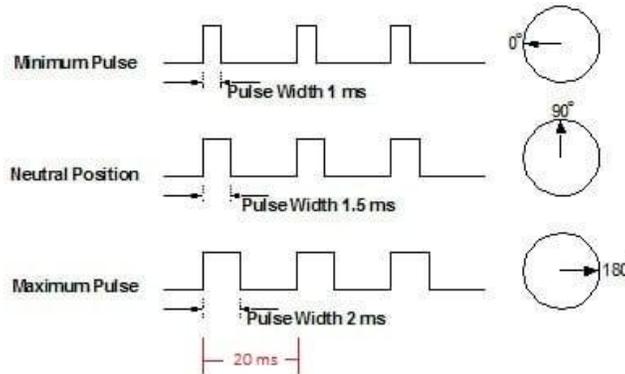


Figura 71. Tiempos de trabajo pwm para un servo motor
Fuente: (Nuñez, 2020)

En el mercado es posible encontrar varios tipos de servos, y es importante entender sus características para adaptarlos adecuadamente a nuestros proyectos, por lo tanto, es de extrema importancia de que leas la hoja de datos del servo que deseas comprar para ver si se adapta en fuerza y grados de movimiento a tu proyecto.

Existen 2 servos básicos que son:

1. Servomotor de -90 grados a 90 grados (total de 180 grados)
 2. Servomotor de 360 grados
- Con el servomotor de 180 grados con Arduino podremos controlar la posición donde deseamos ubicar el motor, por otro lado cuando conectamos un servomotor 360 grados en Arduino ya NO tendremos control de posición, porque el servo gira constantemente. Este servomotor de 360 grados es utilizado para realizar control de velocidad, bastante empleado en pequeños o medianos robots para el control de velocidad de las ruedas.
 - Un factor a tener en cuenta es observar la fuerza del servo, que sea adecuada para mover la carga que necesitamos en nuestro proyecto.

3.1.2 Servo Arduino: Programar Servomotor en Arduino

Vamos a emplear la librería del Arduino llamada Servo Library con la cual controlaremos estos dispositivos de una manera adecuada que nos permite controlar hasta 12 servos en Arduino Uno/Nano, y hasta 48 en Arduino Mega. Claro también puedes controlar el servomotor con Arduino sin Librería y para eso te invito a ver la entrada que hicimos con PIC donde hicimos algo parecido, donde básicamente debemos generar la señal PWM.

El servomotor Puede ser conectado en cualquier PIN DIGITAL de la PLACA DE ARDUINO.

Primero incluimos la librería

```
#include<Servo.h>
```

Definimos el PIN de control donde conectaremos el SERVO

Sintaxis:

```
servo.attach(pin)  
servo.attach(pin, min, max)
```

Parámetros:

servo: una variable de tipo Servo

pin: el número del pin que el servo está conectado

min (opcional): el ancho de pulso, en microsegundos, correspondiente al ángulo mínimo (0 grados) del servo (por defecto 544)

max (opcional): el ancho de pulso, en microsegundos, correspondiente al ángulo máximo (180 grados) del servo (por defecto 2400)

Para escribir en el servo una posición específica usamos el comando:

```
servo.write(angle)
```

Parámetros:

servo: una variable de tipo Servo

angle: el valor a escribir en el servo, de 0 a 180

Para leer la posición actual en la que se encuentra nuestro servo con Arduino podemos usar:

```
servo.read()
```

Parámetros:

servo: una variable de tipo Servo

Retornos:

El ángulo del servo, de 0 a 180 grados

3.1.3 Implementación de control de un grado de libertad simple (1GDL)

1.- Implementación

En base al circuito mostrado en la siguiente figura implementar de acuerdo al orden establecido en los pines del Arduino

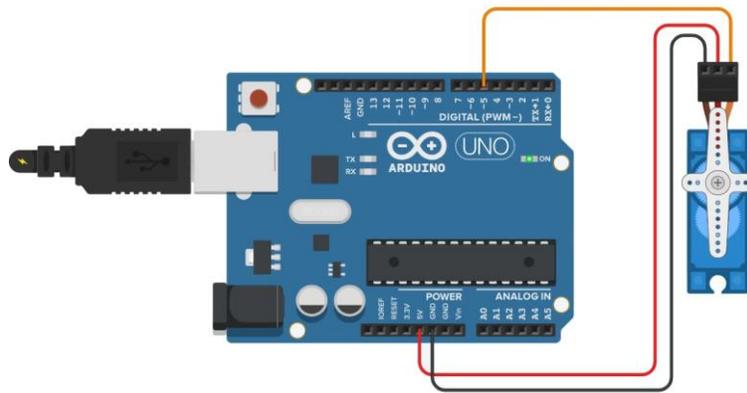


Figura 72. Esquema de conexión control de un grado de libertad simple
Fuente: (Nuñez, 2020)

2.- Programa

Con el siguiente código se pretende tener control en el movimiento de un grado de libertad del servo motor

A.- Control de 0° a 180° y de 180° a 0°

```
#include <Servo.h>

Servo myservo;
int pos = 0;
void setup() {
  myservo.attach(9);
}
void loop() {
  for (pos = 0; pos <= 180; pos += 1) {
    // in steps of 1 degree
    myservo.write(pos);
```

```

delay(15);
}
for (pos = 180; pos >= 0; pos -= 1) {
  myservo.write(pos);
delay(15);
}
}

```

B.- Control con tren de pulsos max y minimos

```

#include <Servo.h>
Servo servo1;
int PINSERVO = 2;          // pin 2 conectado a señal del servo
int PULSOMIN = 1000;      // pulso minimo en microsegundos
int PULSOMAX = 2000;     // pulso maximo en microsegundos
void setup()
{
  servo1.attach(PINSERVO, PULSOMIN, PULSOMAX); // inicializacion de servo
}
void loop()
{
  servo1.write(0);        // ubica el servo a 0 grados
  delay(5000);            // espera de 5 seg.
  servo1.write(180);     // ubica el servo a 180 grados
  delay(5000);           // espera de 5 seg.
}

```

3.- Funciones a observar

- a.- Observar el funcionamiento del recorrido con el servo motor, sentido de giro y la velocidad del mismo.
- b.- Incluir un servo motor para poder controlar en paralelo a los dos dispositivos

3.1.4 Implementación de control de un grado de libertad mediante potenciómetro (1GDL)

1.- Implementación

Dado el siguiente esquema implementar el circuito con la ayuda de un proto board

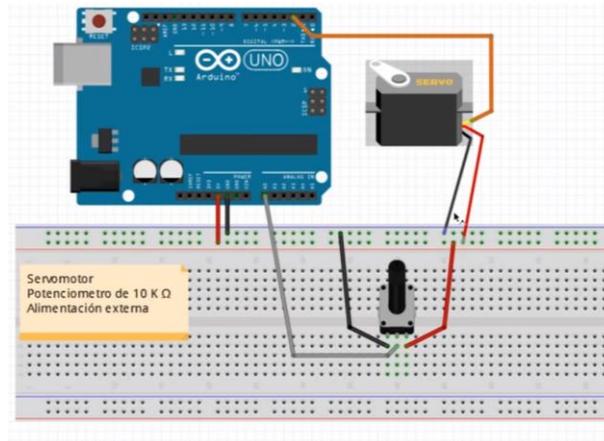


Figura 73. Esquema de conexión control de un grado de libertad mediante potenciómetro
Fuente: (Nuñez, 2020)

2.- Programa

```
#include <Servo.h>           // incluye libreria de Servo
Servo servo1;               // crea objeto
int PINSERVO = 2;          // pin 2 conectado a señal del servo
int PULSOMIN = 1000;      // pulso minimo en microsegundos
int PULSOMAX = 2000;     // pulso maximo en microsegundos
int VALORPOT;              // variable para almacenar valor leído en entrada A0
int ANGULO;                // valor de angulo a cargar en el servo
int POT = 0;               // potenciómetro en entrada analogica A0
void setup()
{
  servo1.attach(PINSERVO, PULSOMIN, PULSOMAX); // inicializacion de servo
  // las entradas analogicas no requieren inicializacion
}
void loop()
{
```

```

VALORPOT = analogRead(POT);           // lee valor de entrada A0
ANGULO = map(VALORPOT, 0, 1023, 0, 180); // con funcion map convierte rango de 0 a
1023

                                        // a rango de angulo de 0 a 180

servo1.write(ANGULO);                 // envia al servo el valor del angulo
delay(20);                             // demora para que el servo llegue a posicion
}

```

3.- Funciones a observar

- Observar el tipo de señal que ingresa al Arduino si es digital o analógica.
- Verificar el funcionamiento de la función map dentro del programa.

3.1.5 Implementación de control de un grado de libertad mediante pulsadores (1GDL)

1.- Implementación

Dado el siguiente esquema electrónico, se pide realizar las conexiones en base a como se indica en la misma considerando el orden de los pines de Arduino.

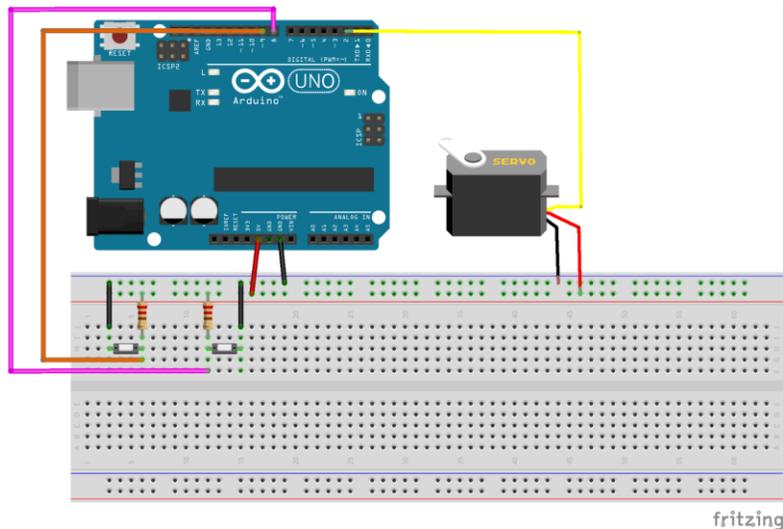


Figura 74. Esquema de conexión control de un grado de libertad mediante pulsadores
Fuente: (Nuñez, 2020)

2.- Programa

```
#include <Servo.h>
```

```
Servo mi_servo;
```

```
int grados = 90; // Inicializa en la posición 90°
```

```
int suma=8; // Boton para sumar angulo
```

```
int resta=9; // Boton para restar angulo
```

```
void setup() {
```

```
    mi_servo.attach(2,750,1800); // Configura el Servo, recuerden que en mi caso 750ms = 0° y 1800 ms = 180°
```

```
    pinMode(suma, INPUT); // Configuramos pines de entrada
```

```
    pinMode(resta, INPUT);
```

```
    mi_servo.write(grados); // Muevo el motor hasta 90°
```

```
    //digitalWrite(suma,HIGH);
```

```
    // digitalWrite(resta,HIGH);
```

```
}
```

```
void loop() {
```

```
    if (digitalRead(suma) == LOW) // Pregunto por el boton suma = 12 Presionado, se activa con estado Bajo
```

```
    {
```

```
        grados++; // Suma grados
```

```
        if (grados >= 180) // Proteje el motor, para que no exceda los 180°, se puede dañar el motor
```

```
        {
```

```
            grados = 180;
```

```
        }
```

```
    }
```

```

if (digitalRead(resta) == LOW)
{
  grados--; // Resta grados
  if (grados <= 0) // Proteje el motor, para que no descienda de los 0°, se puede dañar el motor
  {
    grados = 0;
  }
}
mi_servo.write(grados); // Escribo nuevo angulo en grados

delay(10);
}

```

3.- Funciones a observar

- a.- Como se realiza el incremento de los grados de libertad, describa la función del programa.
- b.- Como se realiza el decremento de los grados de libertad, describa la función del programa.
- c.- Cuan debería ser el esquema de conexión para un control de dos grados de libertad?
- d.- re programe el mismo en base a dos grados de libertad utilizados.

3.1.6 Implementación de control de dos grado de libertad mediante pulsadores (2GDL)

1.- Programa. -

```

#include <Servo.h>

Servo mi_servo1;
Servo mi_servo2;

int grados = 90; // Inicializa en la posición 90°
int posicion = 90;
int suma=8; // Boton para sumar angulo

```

```

int resta=9; // Boton para restar angulo
int aumenta=10;
int disminuye=11;
void setup() {

    mi_servo.attach(2,750,1800); // Configura el Servo, recuerden que en mi caso 750ms = 0° y 1800 ms
    = 180°
    pinMode(suma, INPUT); // Configuramos pines de entrada
    pinMode(resta, INPUT);
    mi_servo.write(90); // Muevo el motor hasta 90°

    mi_servo2.attach(3,750,1800); // Configura el Servo, recuerden que en mi caso 750ms = 0° y 1800
    ms = 180°
    pinMode(aumenta, INPUT); // Configuramos pines de entrada
    pinMode(disminuye, INPUT);
    mi_servo.write(posicion);
    //digitalWrite(suma,HIGH);
    // digitalWrite(resta,HIGH);
}
void loop() {

    if (digitalRead(suma) == LOW) // Pregunto por el boton suma = 12 Presionado, se activa con estado
    Bajo
    {
        grados++; // Suma grados

        if (grados >= 180) // Proteje el motor, para que no exceda los 180°, se puede dañar el motor
        {
            grados = 180;
        }
    }
    if (digitalRead(aumenta) == LOW) // Pregunto por el boton suma = 12 Presionado, se activa con
    estado Bajo

```

```
{  
  posicion++; // Suma grados  
  
  if (posicion >= 180) // Proteje el motor, para que no exceda los 180°, se puede dañar el motor  
  {  
    posicion = 180;  
  }  
}  
  
if (digitalRead(resta) == LOW)  
{  
  grados--; // Resta grados  
  if (grados <= 0) // Proteje el motor, para que no descienda de los 0°, se puede dañar el motor  
  {  
    grados = 0;  
  }  
}  
  
if (digitalRead(disminuye) == LOW)  
{  
  posicion--; // Resta grados  
  if (posicion <= 0) // Proteje el motor, para que no descienda de los 0°, se puede dañar el motor  
  {  
    posicion = 0;  
  }  
}  
  
mi_servo.write(grados); // Escribo nuevo angulo en grados  
  
delay(10);
```

```
lokillo.write(posicion); // Escribo nuevo angulo en grados
```

```
delay(10);
```

```
}
```

2.- Implementación

Dado el código anterior se pide realizar la implementación y control de dos servomotores mediante pulsos.

3.2 Cinemática directa e inversa

Los robots clásicos presentan una arquitectura antropomórfica serial, semejante al brazo humano. Consisten de una serie de barras rígidas unidas entre sí a través de articulaciones de un grado de libertad del tipo rotacional o prismática. En general cada articulación logra su movimiento a través de un accionamiento de potencia e incluye otros dispositivos como reductores de velocidad, frenos y sensores de posición o velocidad (*Práctica 2 Cinemática de Robots*, s. f.).

Aunque al definir las relaciones cinemáticas de un robot no se suelen considerar los aspectos dinámicos, nada más alejado de la realidad cuando se quiere diseñar un robot ya que existe una inevitable relación causa-efecto entre la cinemática y la dinámica. Nada más claro resulta que al pensar en las dimensiones de un robot, la longitud de un brazo afecta al cuadrado la inercia de los eslabones y por lo tanto el peso del robot y la potencia requerida en los actuadores (*Práctica 2 Cinemática de Robots*, s. f.).

3.2.1 Cinemática directa del brazo de un robot manipulador

Las técnicas que se estudian aquí, se aplican a un manipulador mecánico de cadena abierta y tratan el estudio analítico y el modelado en MatLab de la geometría del movimiento de un robot con respecto a un sistema de referencia fijo como una función del tiempo sin considerar la dinámica (*Práctica 2 Cinemática de Robots*, s. f.).

El problema cinemático directo se plantea en términos de encontrar una matriz de transformación que relaciona el sistema de coordenadas ligado al cuerpo en movimiento respecto a un sistema de coordenadas que se toma como referencia. Para lograr esta representación se usan las matrices de transformación homogénea 4x4, la cual incluye las operaciones de traslación y la orientación. La matriz de transformación homogénea es una matriz de 4x4 que transforma un vector expresado en coordenadas homogéneas desde un sistema de coordenadas hasta otro sistema de coordenadas. Para una descripción más amplia acerca de las bases algebraicas de las transformaciones homogéneas (*Práctica 2 Cinemática de Robots*, s. f.).

La matriz de transformación homogénea tiene la siguiente estructura:

$$T = \begin{bmatrix} \text{matriz de rotación} & \text{vector de posición} \\ \mathbf{f}_{1 \times 3} & \text{escalado} \end{bmatrix} = \begin{bmatrix} \mathbf{n}_x & \mathbf{s}_x & \mathbf{a}_x & \mathbf{p}_x \\ \mathbf{n}_y & \mathbf{s}_y & \mathbf{a}_y & \mathbf{p}_y \\ \mathbf{n}_z & \mathbf{s}_z & \mathbf{a}_z & \mathbf{p}_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T = \begin{bmatrix} \mathbf{n} & \mathbf{s} & \mathbf{a} & \mathbf{p} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 75. Matriz de transformación homogénea
Fuente: (*Práctica 2 Cinemática de Robots*, s. f.)

donde los vectores \mathbf{n} , \mathbf{s} , \mathbf{a} , son vectores ortogonales unitarios y \mathbf{p} es un vector que describe la posición x, y, z del origen del sistema actual respecto del sistema de referencia. Para entender las propiedades de la matriz de transformación homogénea nos fijamos en el siguiente gráfico.

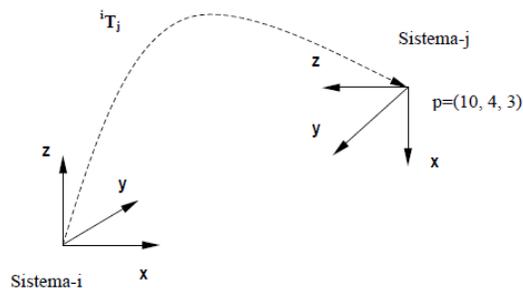


Figura 75. Interpretación geométrica de la matriz de transformación homogénea
Fuente: (*Práctica 2 Cinemática de Robots*, s. f.)

En base a la anterior figura se puede obtener la siguiente matriz

$${}^i\mathbf{T}_j = \begin{bmatrix} 0 & 0 & -1 & 10 \\ 0 & -1 & 0 & 4 \\ -1 & 0 & 0 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Al analizar las columnas de la submatriz de rotación de la matriz de transformación homogénea T_j^i , un observador localizado en el origen de sistema-i, puede ver como están orientados los ejes x, y, z del sistema-j, además también observa como se ha desplazado en coordenadas cartesianas el origen del sistema-j respecto del origen del sistema de referencia con la información del vector de posición (*Práctica 2 Cinemática de Robots*, s. f.).

3.2.2 Cinemática inversa del brazo de un robot manipulador

La cinemática inversa consiste en hallar los valores de las coordenadas articulares del robot $q = [q_1, q_2, \dots, q_n]^T$ conocida la posición y orientación del extremo del robot.

A pesar de que en la literatura se pueden encontrar diversos métodos genéricos para la resolución de la cinemática inversa que pueden ser implementados en computadora, suele ser habitual la resolución por medio de métodos geométricos. La mayor parte de los robots suelen tener cadenas cinemáticas relativamente sencillas, que facilitan la utilización de los métodos geométricos (*Práctica 2 Cinemática de Robots*, s. f.).

Para muchos robots, si se consideran sólo los tres primeros grados de libertad, se tiene una estructura planar. Este hecho facilita la resolución del problema. Asimismo, los últimos tres grados de libertad suelen usarse para la orientación de la herramienta, lo cual permite una resolución geométrica desacoplada de la posición de la muñeca del robot y de la orientación de la herramienta. En esta sección se va a resolver el problema cinemático inverso para los dos robots anteriores, utilizando el método geométrico e implementándolo en Matlab (*Práctica 2 Cinemática de Robots*, s. f.).

3.3 Programación de servomecanismos

Para este apartado se podrá verificar la programación tanto para servomotores, como motores paso a paso, que ayuden a realizar desplazamientos de mecanismos simples.

3.3.1 Programación de la lectura escritura de un servo mecanismo mediante un servo motor

En este apartado lo que se pretende es utilizar al potenciómetro como un sensor de posición para que me pueda leer la posición del servo motor, adaptándolo al eje del mismo como tipo encoder, se usará la comunicación serial del IDE de Arduino para poder leer la posición que nosotros como usuarios lo enviamos al servo motor.

1.- Implementación

Implementar el siguiente sistema, en el mismo se deberá fijar el potenciómetro al eje del servomotor con los acoples que por defecto vienen en el servo

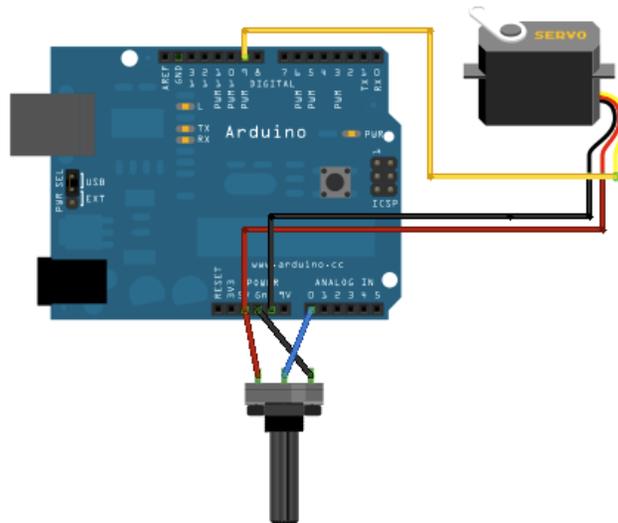


Figura 76. Circuito de lectura y escritura mediante acople de potenciómetro a eje del servo motor
Fuente: (Nuñez, 2020.)

2.- Programación

```

#include <Servo.h>
Servo myservo;
int i=0;
void setup() {

    myservo.attach(7); //
    Serial.begin(9600);
    myservo.write(0);
}
void loop() {
    while(i<=4) {
        myservo.write(0);
        delay(4000);
        float sensorValueD= analogRead(A0); // lee la entrada sobre el pin 0 Analógica
        Serial.println(" ..... ");
        Serial.println("Lectura de A0 en Pos -- 0 -- ");
        Serial.println(sensorValueD); // imprime a la salida el valor de su lectura sobre el pin 0
        delay(1);

        myservo.write(45);//.....(cambia
        el agulo y mira en el puerto serial cuanto lea A0 anlaohamente)
        delay(4000);
        sensorValueD= analogRead(A0); // lee la entrada sobre el pin 0 Analógica
        Serial.println(" ..... ");
        Serial.println("Lectura de A0 en Pos -- 20 --");
        Serial.println(sensorValueD); // imprime a la salida el valor de su lectura sobre el pin 0
        delay(1);

        i=i+1;
    }
}

```

3.- Puntos en discusión

- a.- Análisis de como escribir o enviar el dato de posición en grados al servo motor.
- b.- Análisis de como leer o recibir el dato de posición en la que se encuentra el servo motor

3.3.2 Programación de la escritura y lectura de un servo mecanismo en lazo cerrado, mediante la función escalonamiento y la función map.

1.- Implementación

Implementar el circuito de la figura 56.

2.- Programación

Antes de programar considerar el ángulo a controlar, para lo cual se necesita analizar la función de escalonamiento como se muestra, en nuestro caso trabajaremos con un control de 20°:

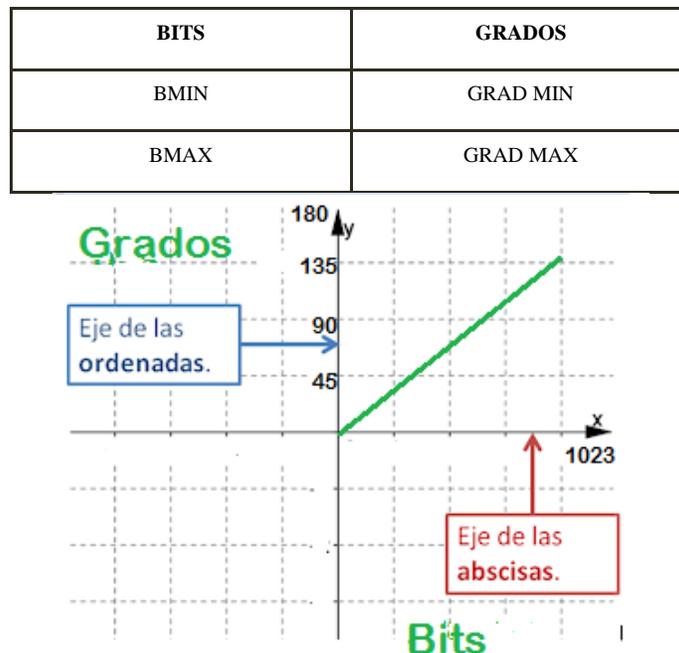


Figura 77. Análisis de conversión entre BITS y GRADOS
Fuente: (Nuñez, 2020.)

$$\bullet \quad y = \frac{Y_{max} - Y_{min}}{X_{max} - X_{min}} * (x - X_{min}) + Y_{min}$$

$$\bullet \quad y = \frac{Grad \ Max - Grad \ Min}{B_{max} - B_{min}} * (In(Ao) - B_{min}) + G_{Min}$$

$$position2 = \frac{20-0}{79-0} * \frac{(sensorValueD)-0}{+0} + 0 = 0,25 * (sensorValueD)$$

$$position2 = 0,25 * (sensorValueD)$$

```

int estado = 0;      //E guarda el estado del botón
int estadoAnterior = 0; //E guarda el estado anterior del botón
int salida = 0;      //E 0 = led pagado, 1 = led encendido
int i=0;
float position;
float position2;

void loop() {

while(i<=4) {
  myservo.write(0);//.....
  delay(4000);
  float sensorValueD= analogRead(A0); // lee la entrada sobre el pin 0 Analogica
  Serial.println(" ..... ");
  Serial.println("Lectura de A0 en Pos -- 0 -- ");
  Serial.println(sensorValueD); // imprime a la salida el valor de su lectura sobre el pin 0

  position = map(sensorValueD, 0, 68, 0, 20);
  Serial.println(" Lectura de A0 en ---0 Grados-- ");
  Serial.println(position); // imprime a la salida el valor de su lectura sobre el pin 0
  Serial.println(" ..... ");
  position2 =0.25*sensorValueD ;
  Serial.println(" funcion escalonamiento " );
  Serial.println(position2);

  delay(1);          // waits 15ms for the servo to reach the position

```

```

myservo.write(20);//.....(cambia el agulo y
mira en el puerto serial cuanto lea A0 anlahamente)
delay(4000);
sensorValueD= analogRead(A0); // lee la entrada sobre el pin 0 Analogica
Serial.println(" ..... ");
Serial.println("Lectura de A0 en Pos -- 20 --");
Serial.println(sensorValueD); // imprime a la salida el valor de su lectura sobre el pin 0

position = map(sensorValueD, 0, 68, 0, 20);
Serial.println("A0 en --- 20 Grados-- ");
Serial.println(position); // imprime a la salida el valor de su lectura sobre el pin 0
Serial.println(" ..... ");
position2 =0.25*sensorValueD ;
Serial.println("funcion escalonamiento ");
Serial.println(position2);
delay(1); // waits 15ms for the servo to reach the position

i=i+1;
}

}

```

3.- Puntos a discusión

- a. Como se utiliza la función map?
- b.- Como funciona la función de escalonamiento?
- c.- Que diferencia tiene la función map con la del escalonamiento?

3.4 Programación de un brazo manipulador

Mediante la utilización de Matlab y la librería de Arduino , se plantea manipular un robot de 3 grados de libertad.

Link: documentos e instaladores plugin para enlazar Matlab con Arduino

http://www.mediafire.com/file/tavhq6wpz54l5m7/Enlace_ARDUINO_-_MATLAB.rar/file

link video descripción de la inflación

<https://www.youtube.com/watch?v=IyeKXNQ6Kww>

Para realizar la primera actividad primero vamos a realizar pruebas de escritura realizando los siguientes comandos en el Comand Window

```
clear all
clear all port
a= arduino('com15') //dependiendo del puerto de su computadora
s=servo(a, 'D8') // Pin digital de salida de su Arduino
writePosition(s,0.5) // mueve 90 grados
```

Se procese a dar un movimiento al servo motor que previamente deberá ya estar conectado:

PC---Arduino----Servomotor

```
clear all port%limpiar todos los puertos
a=arduino('COM3')%puerto asignado a la placa Arduino
s=servo(a, 'D2') %servomotor conectado en la salida PWM 2
for idx=0:2%veces que debe ejecutar el programa
    writePosition (s,0.5)%movimiento del servo a 90 grados
    pause(1)%tiempo de espera 1 segundo
    writePosition (s,1)%movimiento del servo a 180 grados
    pause(1)
end
```

Ahora se va a proceder a realizar un movimiento con 2 servomotores para ello vamos a ingresar el siguiente código

```

clear all port%limpiar todos los puertos
a=arduino('COM3')%puerto asignado a la placa Arduino
s=servo(a,'D2') %servo motor conectado en la salida PWM 2
s1=servo(a,'D3') %servo motor conectado en la salida PWM 3
for idx=0:2%veces que debe ejecutar el programa
    writePosition (s,0.5)%movimiento del servo a 90 grados
    pause(1)%tiempo de espera 1 Segundo
    writePosition (s1,0.5)%movimiento del servo1 a 90 grados
    pause(1)
    writePosition (s,1)%movimiento del servo a 180 grados
    pause(1)
    writePosition (s1,1)%movimiento del servo1 a 180 grados
    pause(1)
    writePosition (s,0.222)%movimiento del servo 40 grados
    pause(1)
    writePosition (s1,0.222)%movimiento del servo 40 grados
    pause(1)
end

```

Control de desplazamiento de un brazo manipular con 3GDL

```

clear all
clear all port
a=arduino ('COM3')
base=servo(a,'D7')
hombro=servo(a,'D6')
pinza=servo(a,'D5')
for i=0:2
    writePosition (hombro,0.7215)%hombro toma posición 130 grados
    pause(1)
    writePosition (pinza,0)%abre pinzas
    pause(1)
    writePosition (base,0.97125)%base mueve 180 grados
    pause (2)
    writePosition (hombro,0.444)%hombro baja hasta 80 grados
    pause (1)
end

```

```
writePosition (pinza,0.19425)%Cierra las pinzas
pause (1)
writePosition (hombro,0.7215)%hombro regresa a 130 grados
pause (1)
writePosition (base,0)%base regresa posición inicial
pause (2)
writePosition (hombro,0.444)%hombro baja hasta 80 grados
pause (1)
writePosition (pinza,0)%abre las pinzas
pause (1)
writePosition (hombro,0.7215)%hombro regresa a 130 grados
pause (1)
end
```

ACTIVIDADES UNIDAD 3

Actividad 1: Preguntas

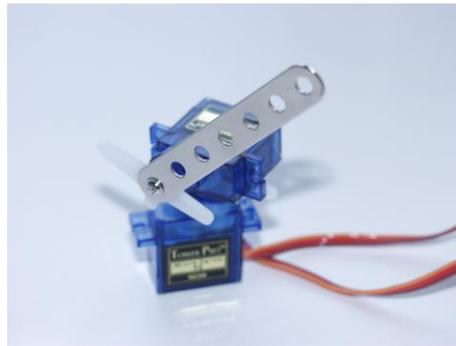
- 1.- Cual es la ecuación para la obtención de GDL en base Gruebler.
- 2.- Cual es la diferencia entre la ecuación de Gruebler y la modificada kutzbach
- 3.- Con que tipo de señal trabajan los servos motores?
- 4.- Cual es el tiempo aproximado que toma la señal del un acho de pulso pwm para poder posicionar a un servo motor a 45º?
- 5.- cuando se utiliza un servo de control de 180º y cuando un servo a 360º?
- 6.- Para poder programar sobre el IDE de Arduino con servo motores , que librería debo incluir?
- 7.- Para definir el pin de control donde conectaremos el servo , qué significado tiene los parámetros de la función a utilizar.
- 8.- Par escribir en el servo una posición con la función *servo.write* que tipo dato debemos proporcionar dentro de la función?
- 9.- Como trabaja la función **map** en el IDE de Arduino, mencione su función con una entrada a controlar de 20 grados con una señal en bits de 500.
- 10.- Detalle cómo se puede hacer una función incremental de acumulación, para el control de una posición por pulsadores.
- 11.- En que consiste la cinemática directa de un brazo manipular?
- 12.- En que consiste la cinemática inversa de un brazo manipular?
- 13.- Describa de consiste la matriz de trasformación homogénea
- 14.- Realice la interpretación geométrica de la matriz de transformación homogénea.
- 15.- Que se necesita para poder escribir una posición en un servo motor y comprobar que dicha escritura logro posicionar en el ángulo correcto.

Actividad 2: Actividades de simulación/ prácticas.

1.- Se desea dar implementación al control de dos servos motores como indica la figura, utilizando un control:

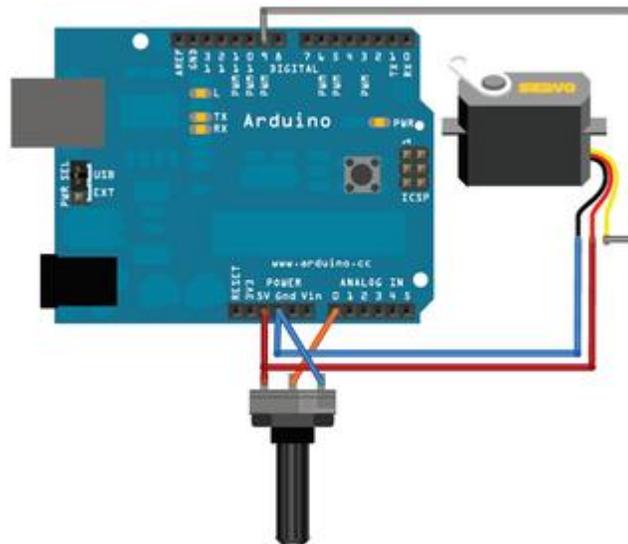
Con dos potenciómetros, uno por cada GDL

Con dos pulsadores, uno por cada GDL



Fuente: (Autores)

2.-Realizar un sistema de control de lazo cerrado para comprobar la posición del servo motor, utilizando un potenciómetro como sensor de retroalimentación y la función de escalonamiento. La imagen adjunta le servirá como referencia de conexión, se podrá realizar como mejor se convenga el acople entre el potenciómetro y el eje del servo motor



Fuente: (Autores)

UNIDAD 4: ROBÓTICA MÓVIL

4.1 Locomoción

Existe una gran variedad de modos de moverse sobre una superficie sólida; entre los robots, las más comunes son las ruedas, las cadenas y las patas.

Los vehículos de ruedas son, con mucho, los más populares por varias razones prácticas. Los robots con ruedas son más sencillos y fáciles de construir, la carga que pueden transportar es mayor, relativamente. Tanto los robots basados en cadenas como en patas se pueden considerar más complicados y pesados, generalmente, que los robots de ruedas para una misma carga útil. A esto podemos añadir el que se pueden transformar vehículos de ruedas de radio control para usarlos como bases de robots (González, 2002a).



Figura 78. Ejemplo de un robot móvil
Fuente: (González, 2002a)

La principal desventaja de las ruedas es su empleo en terreno irregular, en el que se comportan bastante mal. Normalmente un vehículo de ruedas podrá superar un obstáculo que tenga una altura no superior al radio de sus ruedas, entonces una solución es utilizar ruedas mayores que los posibles obstáculos a superar; sin embargo, esta solución, a veces, puede no ser práctica (González, 2002a).

Para robots que vayan a funcionar en un entorno natural las cadenas o orugas son una opción muy buena porque las cadenas permiten al robot superar obstáculos relativamente mayores y son menos susceptibles que las ruedas de sufrir daños por el entorno, como piedras o arena. El principal inconveniente de las cadenas es su ineficacia, puesto que se produce deslizamiento sobre el terreno al avanzar y al girar. Si la navegación se basa en el conocimiento del punto en que se encuentra el

robot y el cálculo de posiciones futuras sin error, entonces las cadenas acumulan tal cantidad de error que hace inviable la navegación por este sistema. En mayor o menor medida cualquiera de los sistemas de locomoción contemplados aquí adolece de este problema (González, 2002a).

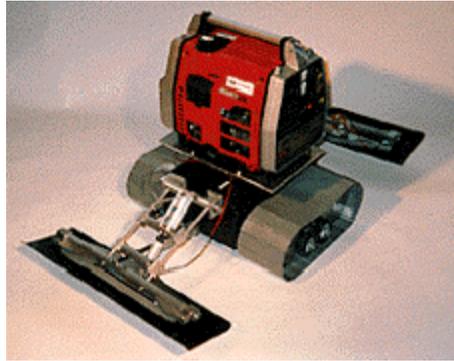


Figura 79. Ejemplo de un robot móvil tipo cadena u oruga
Fuente: (González, 2002a)

Potencialmente los robots con patas pueden superar con mayor facilidad que los otros los problemas de los terrenos irregulares. A pesar de que hay un gran interés en diseñar este tipo de robots, su construcción plantea numerosos retos. Estos retos se originan principalmente en el gran número de grados de libertad que requieren los sistemas con patas. Cada pata necesita como mínimo un par de motores lo que produce un mayor coste, así como una mayor complejidad y menor fiabilidad. Es más los algoritmos de control se vuelven mucho más complicados por el gran número de movimientos a coordinar, los sistemas de patas son un área de investigación muy activo (González, 2002a).

4.1.1 Consideraciones de diseño

Antes de empezar a construir la estructura de locomoción de un robot móvil se deberá tener presente las siguientes consideraciones: (*Sistemas de Locomoción de robots móviles*, s. f.).

- Maniobrabilidad
- Controlabilidad
- Tracción

- Capacidad de subir pendientes
- Estabilidad
- Eficiencia
- Mantenimiento
- Impacto ambiental
- Consideraciones de 'Navegabilidad'

4.1.2 Tipos de locomoción

Dentro de los principales tipos de construcción para la locomoción de un robot móvil podemos citar: (*Sistemas de Locomoción de robots móviles*, s. f.).

- ✓ Con ruedas/cintas de deslizamiento
 - Diferencial
 - Síncrona
 - Triciclo
 - Ackerman
 - Omnidireccionales
 - Otras
- ✓ Con patas
- ✓ Otros

4.1.3 Definiciones entorno a la locomoción de un robot móvil

Para denominar ciertas partes de un robot móvil se deberá considera previo su definición

Rueda motriz

La que proporciona fuerza de tracción al robot

Rueda directriz:

Ruedas de direccionamiento de orientación controlable.

Ruedas fijas:

Sólo giran en torno a su eje sin tracción motriz.

Ruedas locas o ruedas de castor.

Ruedas orientables no controladas.

En la figura siguiente se puede apreciar de mejor manera las definiciones de las diferentes formas de implementar una rueda

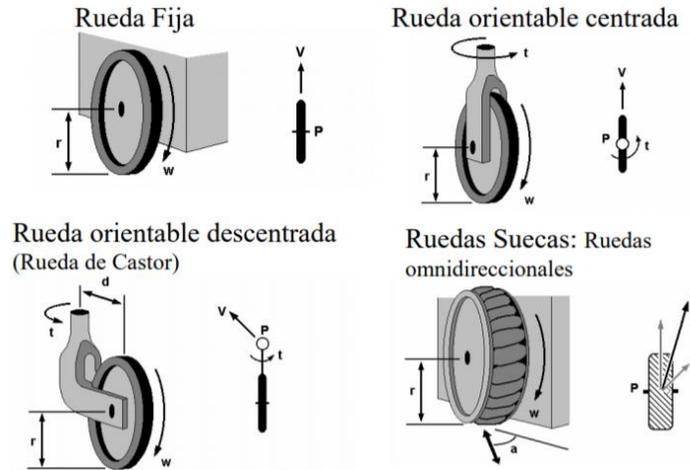


Figura 80. Representación de tipos de ruedas para locomoción en robots móviles
Fuente: (Sistemas de Locomoción de robots móviles, s. f.)

Centro instantáneo de Rotación

(CIR) o centro instantáneo de curvatura (CIC): El punto de intersección de todos los ejes de las ruedas

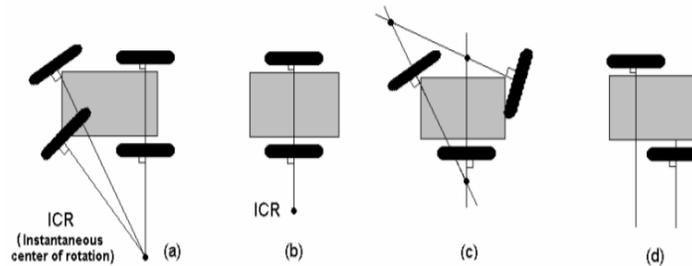


Figura 81. centro instantáneo de curvatura
Fuente: (Sistemas de Locomoción de robots móviles, s. f.)

Restricciones no holónomas

El robot puede moverse instantáneamente adelante o atrás pero no lateralmente por el deslizamiento de las ruedas

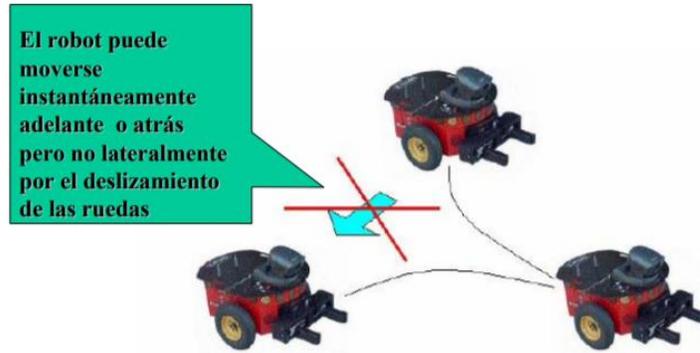


Figura 82. Restricciones no holónomas
Fuente: (Sistemas de Locomoción de robots móviles, s. f.)

Locomoción diferencial

No hay ruedas directrices. El cambio de dirección se realiza modificando la velocidad relativa de las ruedas a Izquierda y Derecha

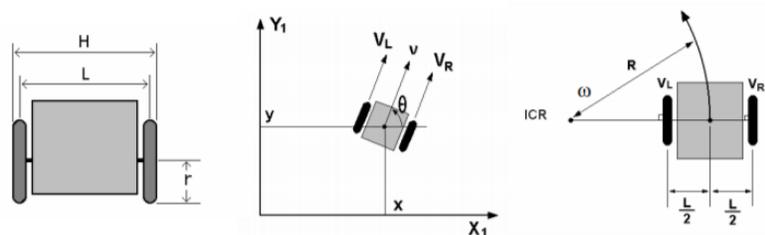


Figura 83. Restricciones no holónomas
Fuente: (Sistemas de Locomoción de robots móviles, s. f.)

Ventajas:

- Sistema Barato
- Fácil de implementar
- Diseño simple

Inconvenientes:

- Difícil de controlar
- Requiere control de precisión para trayectorias rectas

4.1.4 Diseño locomoción por ruedas

Existen varios diseños de ruedas para elegir cuando se quiere construir una plataforma móvil sobre ruedas: diferencial, sincronizada, triciclo y de coche (González, 2002a).

4.1.4.1 Diferencial

Tanto desde el punto de vista de la programación como de la construcción, el diseño diferencial es uno de los menos complicados sistemas de locomoción. El robot puede ir recto, girar sobre sí mismo y trazar curvas (González, 2002a).

Un problema importante es cómo resolver el equilibrio del robot, hay que buscarles un apoyo adicional a las dos ruedas ya existentes, esto se consigue mediante una o dos ruedas de apoyo añadidas en un diseño triangular o romboidal. El diseño triangular puede no ser suficiente dependiendo de la distribución de pesos del robot, y el romboidal puede provocar inadaptación al terreno si éste es irregular lo que puede exigir alguna clase de suspensión (González, 2002a).

Otra consideración a hacer en este diseño es cómo conseguir que el robot se mueva recto, para que el robot se mueva en línea recta sus ruedas tienen que girar a la misma velocidad (González, 2002a).

Cuando los motores encuentran diferentes resistencias (una rueda sobre moqueta y la otra sobre terrazo) las velocidades de los motores varían y el robot girará incluso aún cuando se le haya ajustado inicialmente para que vaya recto. Esto quiere decir que la velocidad debe ser controlada dinámicamente, o sea, debe existir un medio de monitorizar y cambiar la velocidad del motor mientras el robot avanza. De esta manera la simplicidad del diseño queda minimizada por la complejidad del sistema de control de la velocidad; no obstante la reducción de la complejidad mecánica en detrimento de la complejidad de la electrónica y del software es frecuentemente una elección más barata y fiable (González, 2002a).

4.1.4.2 Diseño sincronizado

En este diseño todas las ruedas (generalmente tres) son tanto de dirección como motrices, las ruedas están enclavadas de tal forma que siempre apuntan en la misma dirección. Para cambiar de dirección el robot gira simultáneamente todas sus ruedas alrededor de un eje vertical, de modo que la dirección del robot cambia, pero su chasis sigue apuntando en la misma dirección que tenía. Si el robot tiene una parte delantera (es asimétrico) presumiblemente donde se concentran sus sensores, se tendrá que arbitrar un procedimiento para que su cuerpo se oriente en la misma dirección que sus ruedas. El diseño sincronizado supera muchas de las dificultades que plantean el diseño diferencial, en triciclo y de coche, pero a costa de una mayor complejidad mecánica (González, 2002a).

4.1.4.3 Diseño de triciclo y coche

El diseño de coche con sus cuatro ruedas con suspensión proporciona una buena estabilidad, el diseño en triciclo tiene unas prestaciones similares con la ventaja de ser mecánicamente más simple ya que el coche necesita alguna unión entre las ruedas direccionables. En general en estos dos diseños las ruedas direccionables no son motrices, y no es necesario controlar la velocidad de las ruedas para que el robot se mantenga recto (González, 2002a).

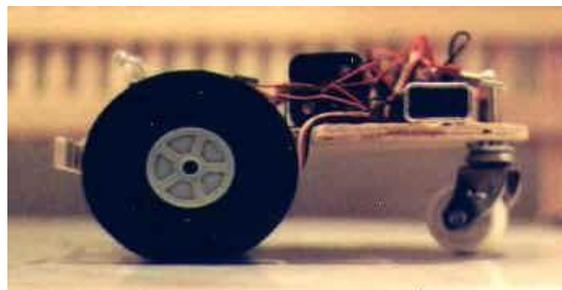


Figura 84. Locomoción robot móvil tipo triciclo, coche
Fuente: (Sistemas de Locomoción de robots móviles, s. f.)

4.1.5 Diseño Locomoción por medio de patas

En general, los sistemas que emplean patas son bastante complejos, sin embargo, hay variantes. Un sistema de patas tipo insecto se puede construir empleando sólo parejas de servos (González, 2002a).

Para dar un paso un servo abre la pata alejándola del cuerpo para salvar si hay un obstáculo, luego el otro servo de la pareja gira para que la pata se mueva adelante. El primer servo, después, baja la pata hasta que ésta toque el suelo, finalmente el segundo servo gira hacia atrás empujando el cuerpo del robot adelante. El movimiento coordinado de seis patas permite al robot moverse adelante, atrás y girar (González, 2002a).

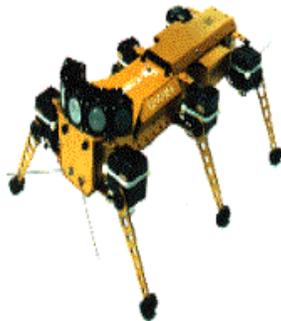


Figura 85. Locomoción por patas
Fuente: (Sistemas de Locomoción de robots móviles, s. f.)

4.2 Cinemática del robot

La cinemática del robot trata de cómo se mueve el mismo, dado que la dirección adopta tal y tal ángulo y que cada rueda gira tantas veces, ¿dónde acabará el robot y qué camino tomará?

Los diseños diferencial y sincronizado tienen una ventaja sutil sobre los otros dos tipos, la diferencia estriba en sus cinemáticas. Considere un triciclo, el cual tiene tres grados de libertad cuando se mueve sobre una superficie plana. Es decir, en relación con un sistema global de coordenadas, el robot puede estar en cualquier posición especificado por dos coordenadas x e y , y apuntando en una dirección especificada por una tercera coordenada, el ángulo q . Estos tres grados de libertad (x,y,q) nos dan la distancia y el ángulo entre el sistema de coordenadas global, y una referencia local en el robot (González, 2002a).

Nos gustaría tener la posibilidad de posicionar y orientar nuestro robot en cualquier lugar sobre el plano, es decir, sin considerar de donde arranca, si le damos x,y,q el robot debe poder moverse a

esa posición. Sin embargo, hay un problema, para alcanzar esos tres grados de libertad el robot sólo puede controlar dos parámetros: la dirección, ángulo α , y la distancia total recorrida, S . Esto quiere decir que la orientación del robot y su posición están ligados, para girar tiene que moverse hacia delante o hacia atrás. El robot no puede ir directamente de una posición y/o orientación a otra, incluso aún cuando no haya nada en su camino (González, 2002a).

Para alcanzar una posición y orientación deseadas simultáneamente, el robot tiene que seguir algún camino, posiblemente complejo. Los detalles de ese camino se complican todavía más con la presencia de obstáculos, esta es la razón por la que el aparcamiento en línea es difícil; sin embargo, un robot basado en los diseños diferencial o sincronizado puede, al girar sobre sí mismo, desacoplar efectivamente su posición de su orientación (González, 2002a).

4.2.1 Cinemática de Robots Móviles

La cinemática de un robot móvil describe la evolución de la posición/orientación del mismo en función de las variables de actuación. Se desarrollará únicamente para robots con ruedas

Hipótesis previas al análisis de una cinemática

- El Robot se mueve sobre una superficie plana
- Los ejes de guiado son perpendiculares al suelo
- Rodadura pura (No hay deslizamiento)
- Robot como sólido rígido (no hay flexión)
- Las trayectorias se pueden aproximar como arcos de circunferencia entre dos periodos de muestre consecutivos.

4.2.2 Modelo básico del monociclo

La cinemática empleada en este tipo de robots móviles, considera si perspectiva, en base a la velocidad angular, lineal del sistema o mecanismo a movilizar, como lo muestra la siguiente imagen

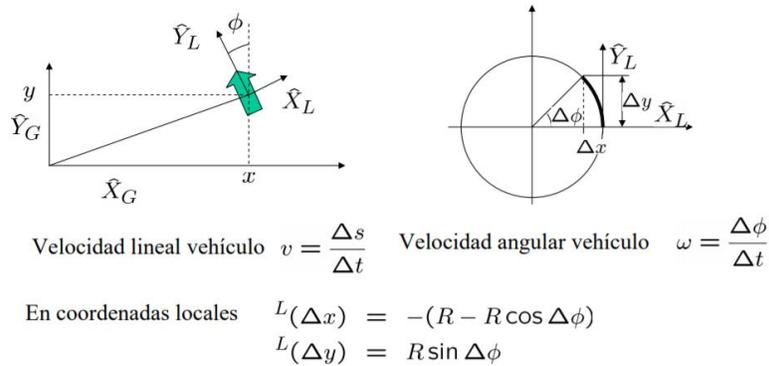


Figura 86. Cinemática Básica del monociclo
 Fuente: (Sistemas de Locomoción de robots móviles, s. f.)

4.2.3 Modelo cinemático Jacobiano

En general la cinemática de un robot móvil puede expresarse aplicando el modelo jacobiano, como muestra la siguiente figura:

$$\dot{p} = J(p)\dot{q} \quad \rightarrow \quad \begin{array}{l} p \text{ Coordenadas generalizadas} \\ q \text{ Variables de actuación} \\ J(p) \text{ Matriz Jacobiana} \end{array}$$

Ej: Para el monociclo

$$\begin{array}{l} p = (x, y, \phi)^T \\ \dot{q} = (v, \omega)^T \end{array} \quad \rightarrow \quad \dot{p} = J(p)\dot{q} \quad \text{con} \quad J(p) = \begin{pmatrix} -\sin \phi & 0 \\ \cos \phi & 0 \\ 0 & 1 \end{pmatrix}$$

Figura 87. Modelo Cinemático Jacobiano
 Fuente: (Sistemas de Locomoción de robots móviles, s. f.)

Permite obtener las variables de actuación necesarias para seguir una determinada trayectoria. Si partimos del modelo Jacobiano podemos obtener el modelo inverso como:

$$\dot{q} = J^*(p)\dot{p} \quad \text{con} \quad J^*(p) = (J^T(p) \cdot J(p))^{-1} J^T(p)$$

4.2.4 Modelos cinemáticos de algunas configuraciones

4.2.4.1 Locomoción Síncrona

Las ruedas se mueven sincrónicamente en velocidad y orientación por lo que el modelo es idéntico al del monociclo, en la siguiente imagen se aprecia la aplicación de la cinemática

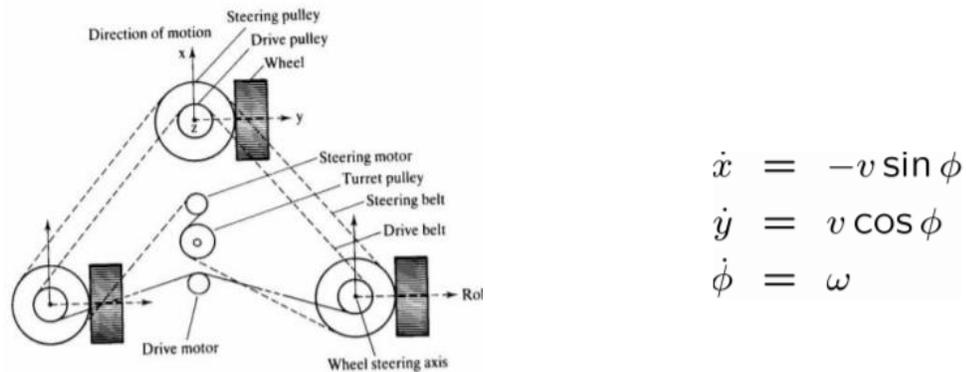


Figura 88. Modelo Cinemático Locomoción sincrona
Fuente: (Sistemas de Locomoción de robots móviles, s. f.)

4.2.4.2 Locomoción Diferencial.

Aplicado a un movimiento de ruedas en un robot móvil, el modelo se reduce al del monociclo con las siguientes ecuaciones:

$$v = \frac{v_L + v_R}{2} = \frac{R(\omega_L + \omega_R)}{2}$$

$$\omega = \frac{v_R - v_L}{L} = \frac{R(\omega_L - \omega_R)}{L}$$

La estructura cinemática será:

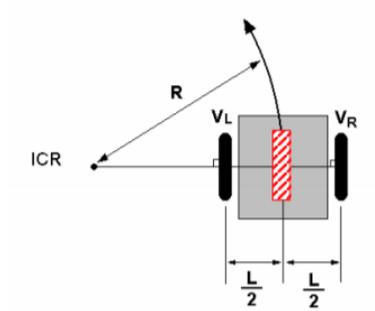


Figura 89. Modelo Cinemático Locomoción Diferencial
Fuente: (Sistemas de Locomoción de robots móviles, s. f.)

4.2.4.3 Triciclo/Bicicleta

Modelo cinemático para robots móviles de 2 a tres ruedas, o rueda giratoria de arrastre.

El modelo se reduce al del monociclo con

$$v = v_t \cos \alpha$$

$$\dot{\alpha} = \omega_\alpha$$

Y el cambio de orientación del vehículo

$$\dot{\phi} = \frac{v_t}{d} \sin \alpha$$

En la siguiente imagen se puede apreciar la cinemática del sistema de locomoción

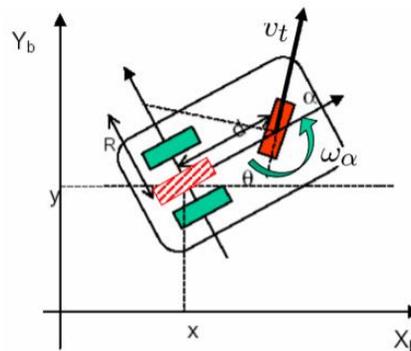


Figura 90. Modelo Cinemático Triciclo/Bicicleta
Fuente: (Sistemas de Locomoción de robots móviles, s. f.)

4.3 Percepción

Los humanos no damos, a menudo, importancia al funcionamiento de nuestros sistemas sensoriales. Vemos una taza sobre una mesa, la cogemos automáticamente y no pensamos en ello, al menos no somos conscientes de pensar mucho en ello. De hecho, el conseguir beber de una taza requiere una compleja interacción de sentidos, interpretación, conocimiento y coordinación, que, en la actualidad, entendemos mínimamente (González, 2002a).

Por tanto, infundir a un robot prestaciones de tipo humano resulta ser tremendamente difícil. Los juegos de ordenador que derrotan a los campeones de ajedrez son comunes en nuestros días, mientras que un programa que reconozca una silla, por ejemplo, en una escena arbitraria aún no existe. El "ordenador paralelo" que todos tenemos en nuestra cabeza dedica grandes cantidades de materia gris a los problemas de la percepción y la manipulación (González, 2002a).

De modo esquemático el proceso de percepción se puede dividir en tres fases (Valero, s. f.).

4.3.1 Fase de Adquisición.

- Depende del tipo de percepción de que se trate.
- Las disciplinas relevantes aquí serán la electrónica y ciertos campos específicos de la Física.
- Su aportación a la siguiente fase será, en general, una representación matricial cuantitativa y digitalizada de la información ambiental adquirida.

4.3.2 Fase de Tratamiento.

- Abarca aspectos muy diversos (corrección de distorsiones, eliminación de ruido, segmentación y extracción de características etc.) de importancia variable en función de: (1º) Tipo, cantidad y calidad de la información suministrada por la 1ª fase, y (2º) Requisitos de la 3ª fase.

- Las técnicas de base utilizadas son, hasta cierto punto, independientes de la naturaleza de la información procesada. En general corresponden al conjunto de métodos reunidos bajo el nombre de "procesado digital de señales".
- La salida de esta fase será una representación simbólica de la información.

4.3.3 Fase de Interpretación.

- Es la de mayor dificultad del proceso.
- Representa, fundamentalmente, la aportación de la I.A. y de las técnicas del Reconocimiento de Formas en su doble vertiente de decisión teórica y sintáctica.
- El resultado será una descripción precisa y completa del entorno de interés.



Figura 91. Diagrama de bloques del proceso de percepción
Fuente: (Valero, s. f.)

4.3.4 Transducción / Entendimiento

A pesar de que nos gustaría que entendiera y fuera consciente de su entorno, en realidad, un robot está limitado por los sensores que le proporcionamos y el programa que le cargamos. Sentir no es percibir. Los sensores son meramente traductores que convierten algunos fenómenos físicos en señales eléctricas que un microprocesador puede leer (González, 2002a).

Esto se podría hacer mediante un convertidor analógico – digital (A/D) en el microprocesador, cargando el valor de un puerto de entrada – salida (I/O), o usando una interrupción externa. Siempre se necesita que haya alguna interfaz electrónica entre el sensor y el microprocesador para acondicionar o amplificar la señal. (González, 2002a)

4.4 Planeación de rutas

La navegación es la técnica de conducir un robot móvil mientras atraviesa un entorno para alcanzar un destino o meta sin chocar con ningún obstáculo.

Cuando se desconoce el entorno, el robot debe poseer la capacidad de reaccionar ante situaciones inesperadas, esto se logra a través de la percepción del entorno mediante el uso de sensores. Mientras que si se trata de un entorno conocido, el uso de los sensores se vuelve secundario y las tareas a seguir serían: planificar una trayectoria óptima libre de obstáculos, a partir de puntos de partida y llegada y obviamente que el robot pueda seguir y cumplir físicamente esta trayectoria (Yandún & Sotomayor, s. f.).

Se considera a un robot como un objeto rígido al cual se le puede asociar un sistema de coordenadas móvil: $q = (p, \theta)$.

Donde p es la posición y θ la orientación. La localización del vehículo en un determinado instante de tiempo se encuentra definida por la relación existente entre el sistema de coordenadas global F_g en virtud del cual está definido todo el entorno de trabajo y su sistema de coordenadas locales asociado F_r

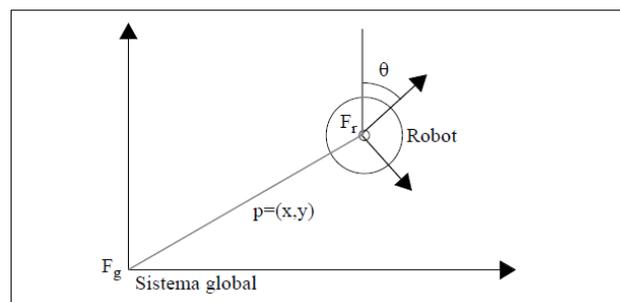


Figura 92. Sistema de coordenadas global, y sistema local asociado al robot
Fuente: (Yandún & Sotomayor, s. f.)

4.5 Métodos de planificación de trayectorias

4.5.1 Grafos de visibilidad:

Para la generación del grafo, este método introduce el término de visibilidad, según el cual, define dos puntos del entorno como visibles si y solo si se pueden unir mediante un segmento rectilíneo que no intercepte ningún obstáculo (Yandún & Sotomayor, s. f.).

Se consideran como nodo del grafo a la posición inicial, la posición final y todos los vértices de los obstáculos, siendo el grafo el resultado de la unión de los nodos visibles, tal como se muestra en la figura (Yandún & Sotomayor, s. f.).

A través de un algoritmo de búsqueda de grafos se escoge la ruta más óptima que una la posición inicial con la final.

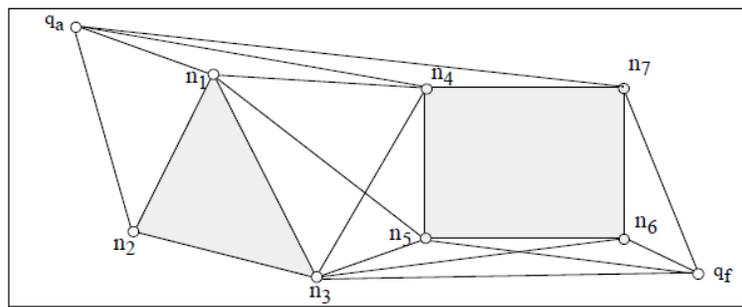


Figura 93. Sistema de coordenadas global, y sistema local asociado al robot
Fuente: (Yandún & Sotomayor, s. f.)

4.5.2 Diagramas de Voronoi:

Se encuentran entre las más importantes estructuras en geometría computacional, este diagrama codifica la información de proximidad entre elementos.

Los diagramas de Voronoi se definen como una proyección del espacio libre del entorno en una red de curvas unidimensionales yacientes en dicho espacio libre. Formalmente se definen como una retracción (Janich, 1984) con preservación de la continuidad. (Yandún & Sotomayor, s. f.)

La principal idea de la construcción del diagrama de Voronoi es ampliar al máximo la distancia que existe entre el robot y los obstáculos, por tanto, el diagrama resulta el lugar geométrico de las configuraciones que se encuentran a la misma distancia de los obstáculos más próximos del entorno (Yandún & Sotomayor, s. f.).

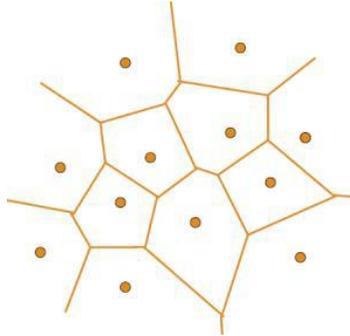


Figura 94. Diagrama de Voronoi de obstáculos representados como puntos.
Fuente: (Yandún & Sotomayor, s. f.)

Para una representación más real se consideran a los obstáculos como polígonos, debido a que físicamente un obstáculo no es un punto. El diagrama de Voronoi estará compuesto por dos tipos de segmentos: rectas y parábolas (Yandún & Sotomayor, s. f.).

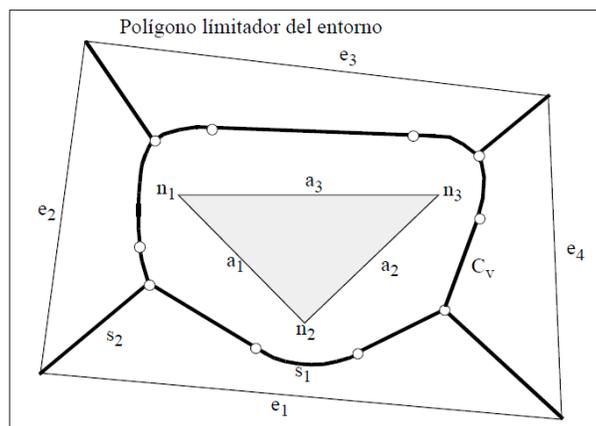


Figura 95. Retracción del espacio libre en un diagrama de Voronoi
Fuente: (Yandún & Sotomayor, s. f.)

El lugar geométrico de las configuraciones que se hallan a la misma distancia de dos aristas de dos obstáculos distintos será una recta, y si se trata de una arista y un vértice, será una parábola (Yandún & Sotomayor, s. f.).

4.5.3 Roadmap Probabilístico (PRM):

Consiste en generar un número n de configuraciones libres de colisión de forma aleatoria y uniforme en toda el área de trabajo.

Se prosigue a conectar cada uno de los nodos con sus nodos más cercanos según una métrica que depende del número de objetos en el entorno de trabajo y finalmente se aplica un algoritmo que obtenga la ruta más óptima, en este caso el algoritmo A*(Yandún & Sotomayor, s. f.).

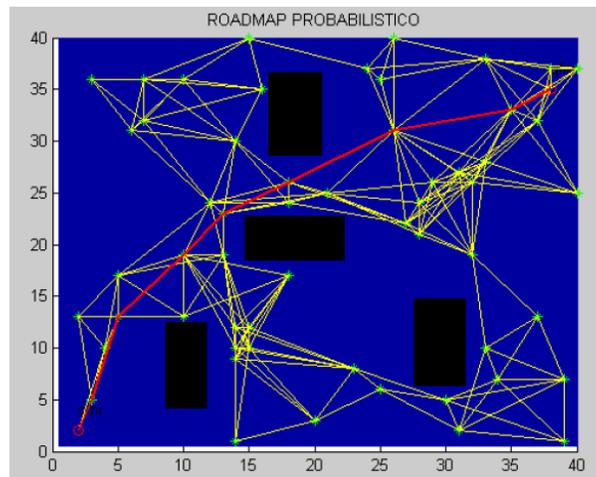


Figura 96. Enlaces válidos en la etapa de conexión y camino mediante el algoritmo A*
Fuente: (Yandún & Sotomayor, s. f.)

4.5.4 Modelado del Espacio Libre:

Igualmente en este método los obstáculos se los representa como polígonos. La planificación se lleva a cabo a través de los CRG, cilindros rectilíneos generalizados, y al igual que Voronoi, con el uso de los CRG se pretende que el robot se mueva lo más alejado de los obstáculos. La ruta será una configuración de CRG interconectados, tal que la configuración inicial o de partida se encuentre en el primer cilindro de la sucesión y la configuración final en el último cilindro.

El proceso para construir un CRG será el siguiente: (Yandún & Sotomayor, s. f.).

- Cálculo del eje del CRG, se define como la bisectriz del ángulo α formado por el corte de las rectas que contienen las aristas $1a_i$ y $2a_j$ que cumplen con las condiciones antes mencionadas (Yandún & Sotomayor, s. f.).
- Por ambos lados de dichas aristas se construyen rectas paralelas al eje, con origen en los vértices de las aristas implicadas y con extremo señalado por la proyección del primer obstáculo que corta el eje.

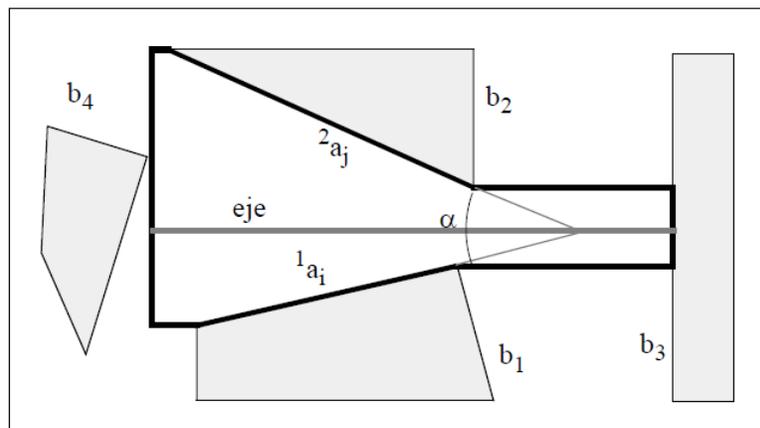


Figura 97. Construcción de un CRG,
Fuente: (Yandún & Sotomayor, s. f.)

Repitiendo este proceso se construye una red CRG en el entorno del robot que modela el espacio libre del mismo. El robot navegará por el eje del cilindro, en el cual se encuentran anotadas para cada punto el rango de orientaciones admisibles (Yandún & Sotomayor, s. f.).

4.5.5 Descomposición en celdas:

En este método no se encuentra una sucesión de segmentos, sino una sucesión de celdas, por lo que es necesaria la construcción de un grafo de conectividad encargado de definir la ruta además de la descomposición de celdas (Yandún & Sotomayor, s. f.).

Dentro de los métodos basados en descomposición en celdas, el método más sencillo es el de descomposición trapezoidal. Se construyen rectas paralelas al eje Y a partir de los vértices de cada

elemento del entorno, estas rectas quedan delimitadas por el corte con las líneas de los elementos del entorno.

El grafo de conectividad se construye a través de la unión de los puntos medios de las rectas definidas, como se observa en la figura 98 (Yandún & Sotomayor, s. f.).

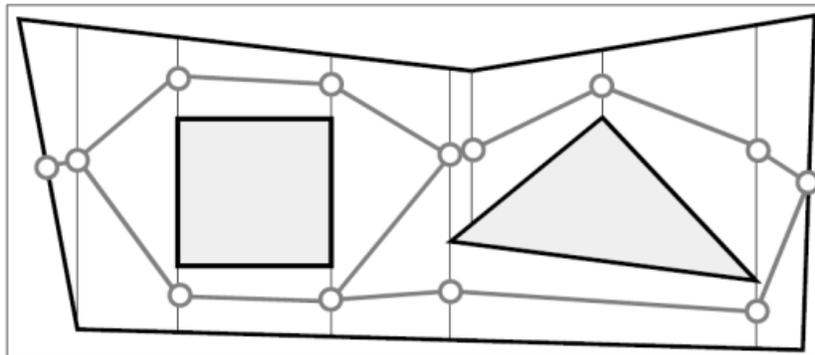


Figura 98. Grafo de conectividad de una descomposición trapezoidal
Fuente: (Yandún & Sotomayor, s. f.)

4.6 Introducción a la Visión Artificial.

Visión artificial o visión por computador es una disciplina científica que incluye métodos para adquirir, procesar, analizar y comprender la realidad de imágenes del mundo con el fin de entregar información numérica o simbólica para que puedan ser entendidos y tratados por un procesador, para luego realizar acciones de acuerdo a propósito general (Nuñez, 2017).

Al igual que los seres humanos usan el proceso de visualizar por medio de los ojos y del cerebro para poder proesar las imágenes observadas, la visión por computador trata de realizar el mismo efecto para que las computadoras puedan percibir y comprender una imagen o secuencia de imágenes y actuar según convenga en una determinada situación utilizando su procesador (Nuñez, 2017).

La visión artificial o visión por computadora está formada por un conjunto de procesos orientados a realizar el análisis o entendimiento de imágenes. Estos procesos son: captación de imágenes, memorización de la información, procesado e interpretación de los resultados (Nuñez, 2017).

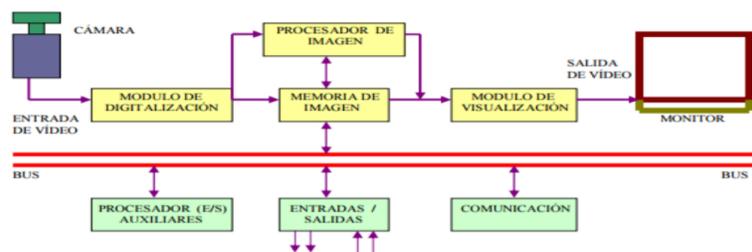


Figura 99. Diagrama de bloques del proceso de la visión artificial
Fuente:(Nuñez, 2017)

Módulo de digitalización: Convierte la señal analógica proporcionada por la cámara a una señal digital (para su posterior procesamiento).

Memoria de imagen: Almacena la señal procedente del módulo de digitalización.

Módulo de visualización: Convierte la señal digital residente en memoria, en señal de vídeo analógica para poder ser visualizada en el monitor de TV. **Procesador de imagen:** Procesa e interpreta las imágenes captadas por la cámara.

Módulo de entradas/salidas: Gestiona la entrada de sincronismo de captación de imagen y las salidas de control que actúan sobre dispositivos externos en función del resultado de la inspección.

Comunicaciones: Vía I/O, ethernet, usb, Rs232, Rs485

4.6.1 Procesos de visión artificial

Una vez entendido el concepto de la visión artificial por computadora se detallará los procesos que hacen posible poder captar y procesar imágenes con fines particulares como son: Adquisición, procesamiento de la imagen, representación y descripción (extracción de características), reconocimiento e interpretación (Nuñez, 2017).



Figura 100. El proceso de la visión artificial
Fuente:(Nuñez, 2017)

4.6.1.1 Adquisición

La adquisición consiste en obtener una representación digital de la imagen de interés dentro del ordenador o controlador. A partir de ese momento será éste el que permita realizar las operaciones sobre dicha imagen para lograr manipular dicha representación digital. La adquisición de la imagen deberá ser lo más clara y estable para poder tener un buen procesamiento y tratamiento de la imagen (Nuñez, 2017).

4.6.1.2 Procesamiento de la Imagen

En esta etapa se procesa la imagen con el objetivo de hacer de la misma una fuente de información, lo más precisa y confiable, dentro de lo que será el proceso de reconocimiento, la calidad del procesador del controlador o computador será clave para poder procesar mucho más rápido esta información. Se divide en los siguientes puntos: binarización, segmentación (Nuñez, 2017).

Binarización: La mayor parte de los algoritmos para el reconocimiento de la escritura están enunciados a partir de imágenes binarias, por lo que se hace conveniente el paso de una imagen en niveles de gris (o color) a una binaria: esto permite, además, reducir el volumen de los datos que serán procesados (Nuñez, 2017).

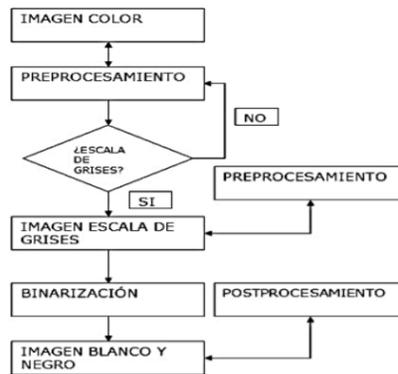


Figura 101. El procesamiento para binarizar una imagen

Fuente:(Nuñez, 2017)

Segmentación: La segmentación es una parte de la procesada digital de imágenes que consiste en la detección de objetos o regiones de interés en ellas. El éxito del proceso depende del paso anterior (Nuñez, 2017).

4.6.1.3 Representación y descripción (extracción de características)

Ya dividida la imagen en zonas con características de más alto nivel se pasará a la extracción de las características o formas a analizar. Básicamente son de tipo morfológico, tales como área, perímetro, excentricidad, momentos de inercia, esqueletos, pero también se puede emplear características basadas en la textura o en el color (Nuñez, 2017).

Nótese que se ha pasado de una información visual primaria o básica a algo más elaborado, se ha obtenido información numérica de una imagen. Con las características analizadas ahora cada región se debe de clasificar e interpretar (Nuñez, 2017).



Figura 102. Extracción de caracteres de una imagen
Fuente:(Nuñez, 2017)

4.6.1.4 Reconocimiento e interpretación

Cuando la imagen ha sido satisfactoriamente segmentada en un alto nivel y han sido extraídas las características necesarias, el procesador puede hacer una serie de pruebas, mediciones y cálculos en los elementos de interés que aparecen en la escena. Existen multitud de algoritmos que nos ayudan a obtener los resultados deseados. Entre ellos, cabe destacar sus funcionalidades:

- Detección de formas
- Análisis geométrico
- Comparación con patrones

- Medición de objetos
- Detección de objetos difusos



Figura 103. Segmentación de una imagen
Fuente:(Nuñez, 2017)

4.6.2 Practicas básicas a la introducción de la visión artificial utilizando Matlab

Para realizar las practicas que a continuación se muestran, se deberá realizar previa la instalación de Matlab. En estas prácticas lo que se necesitara es al menos una cámara web que puede ser de una portátil, o externa. Para ello se empezará desde cómo obtener una imagen, como procesarla y llegar a binarizarla.

4.6.2.1 Lectura de una imagen desde el ordenador

La finalidad de este código a implementar desde Matlab es obtener una lectura desde el mismo ordenador para luego poder realizar un preprocesamiento básico.

```

%Laboratorio visión artificial
%
clear all;
clc;
I=imread('arroz.jpg');
% eSpecificar el archivo de la imagen con su extensión a ser leída, con el mismo nombre
imshow(I);
%se planifica reorganizar el tamaño de escala y colores de la imagen
Isize=imresize(I,0.25);
level= graythresh(I);
Ibin=im2bw(I,level);

figure(1)
subplot(1,3,1);imshow(I);
subplot(1,3,2);imshow(Isize);
subplot(1,3,3);imshow(Ibin);

%figure(2);imshow(Isize);

```

4.6.2.2 Lectura de una imagen desde el ordenador para su binarizarla

En este proceso lo que se pretende es a la imagen anterior realizar el proceso de binarización y mostrar cual es el efecto

```

%%Laboratorio
clear all;
clc;
I=imread('arroz.jpg');
imshow(I);
Isize=imresize(I,0.25);

```

```

level= graythresh(I);
Ibin=im2bw(I,level);
Ibinh=im2bw(I,0.95);
Ibins=im2bw(I,0.1);

figure(1)
subplot(2,2,1);imshow(I);
subplot(2,2,2);imshow(Ibin);
subplot(2,2,3);imshow(Ibinh);
subplot(2,2,4);imshow(Ibins);

%figure(2);imshow(Isize);

```

4.6.2.3 Lectura de una imagen desde el ordenador y manejo de ventanas

Para poder ver la diferencia entre cada procesamiento de imagen con el real es considerable trabajar en varias ventanas de resultados.

```

%%Laboratorio

clear all;
clc;
I=imread('s1.jpg');
J=imread('s3.jpg');
T=I+J;
imshow(T);

Isize=imresize(T,0.25);
level= graythresh(T);

```

```
Tbin=im2bw(T,level);  
Tbinh=im2bw(T,0.95);  
Tbins=im2bw(T,0.1);
```

```
figure(1)  
subplot(2,2,1);imshow(T);  
subplot(2,2,2);imshow(Tbin);  
subplot(2,2,3);imshow(Tbinh);  
subplot(2,2,4);imshow(Tbins);
```

4.6.2.4 Proceso de desfragmentación RGB de la imagen

Para poder empezar a realizar un procesamiento de la imagen es necesario transformar la imagen a una escala de RGB para poder generar una matriz de colores y poder realizar operaciones matriciales con la información obtenida

```
%%Laboratorio DOMOTICA 1
```

```
clear all;
```

```
clc;
```

```
I=imread('s1.jpg');
```

```
J=imread('s3.jpg');
```

```
l=imread('x1.jpg');
```

```
m=imread('x2.jpg');
```

```
%I=T-J;
```

```
imshow(I);
```

```
Igray = rgb2gray(I);
```

```
Isize=imresize(I, 0.25);
```

```
level= graythresh(I);
```

```
Ibin= im2bw(I,level);
```

```
IREd = I; IRed(:,:,2)=0;IREd(:,:,3)=0;
```

```
IGreen=I; IGreen(:,:,1)=0;Green(:,:,3)=0;
```

```
IBlue=I; IBlue(:,:,1)=0;IBlue(:,:,2)=0;
```

```
IgrayR=I(:,:,1);
```

```
IgrayG=I(:,:,2);
```

```
IgrayB=I(:,:,3);
```

```
level=graythresh(IgrayR);IbinR=im2bw(IgrayR,level);
```

```
level=graythresh(IgrayG);IbinG=im2bw(IgrayG,level);
```

```
level=graythresh(IgrayB);IbinB=im2bw(IgrayB,level);
```

figure(1)

subplot(4,3,1);imshow(I);

subplot(4,3,3);imshow(Ibin);

subplot(4,3,4);imshow(IRed);

subplot(4,3,7);imshow(IGreen);

subplot(4,3,10);imshow(IBlue);

subplot(4,3,5);imshow(IgrayR);

subplot(4,3,8);imshow(IgrayG);

subplot(4,3,11);imshow(IgrayB);

%subplot(4,3,6);imshow(IbinR);

subplot(4,3,9);imshow(IbinG);

subplot(4,3,12);imshow(IbinB);

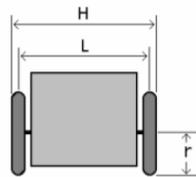
figure (2)

*imshow(I*IbinB);*

ACTIVIDADES UNIDAD 4

Actividad 1: Preguntas

- 1.- ¿Qué se entiende por locomoción en un sistema robótico?
- 2.- ¿Cuál es la desventaja de un robot móvil tipo cadena u oruga?
- 3.- ¿Qué tipo de robot móvil podrá superar fácilmente terrenos irregulares?
- 4.- ¿Qué consideraciones de diseño se Debra tomar en cuenta para implementar un robot móvil que trabajara bajo una superficie de arena?
- 5.- ¿Qué se entiende por ruedas de castor, en un robot móvil?
- 6.- Represente el CIR de un sistema de locomoción de 4 ruedas en una figura.
- 7.- ¿Qué se refiere con restricción no holónomas?
- 8.- ¿Cuáles son las desventajas de ocupar un sistema de locomoción como muestra la imagen?



- 9.- ¿Qué complejidad es la que tiene el diseño sincronizado de locomoción?
- 10.- ¿Cuál es la sincronización de pasos que se debe dar en un movimiento de locomoción bajo el diseño por patas?
- 11.- ¿Qué describe la cinemática de un robot móvil?
- 12.- ¿Cómo se puede obtener la velocidad angular en un modelo básico del monociclo?
- 13.- Dibujar el modelo cinemático de Locomoción diferencial, e identificar la velocidad lineal por formula.
- 14.- ¿Qué se entiende por percepción, en un sistema robótico móvil?
- 15.- ¿Cuáles son las fases de la percepción, detalle cada una?
- 16.- ¿Qué se entiende por planeación de ruta en un robot móvil?
- 17.- Enumere los métodos de planificación de trayectorias.
- 18.- ¿Qué se entiende por visión artificial?
- 19.- ¿Cuál es el proceso que sigue la visión artificial?
- 20.- ¿Qué funciones realiza la etapa de reconocimiento e interpretación?

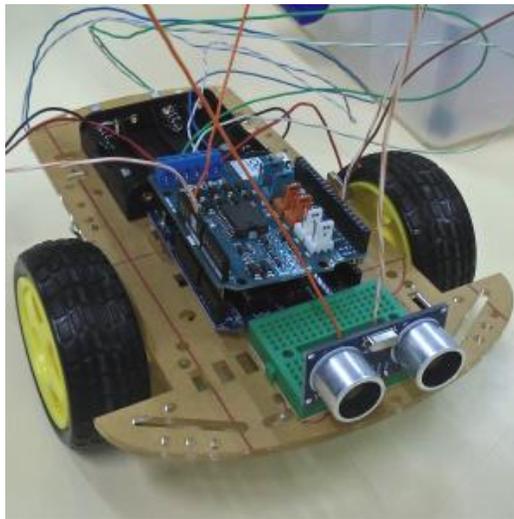
Actividad 2: Actividades de simulación/ prácticas.

1.- Diseñar un robot móvil aplicando una locomoción de rueda loca, con la finalidad de realizar un seguidor de luz, auto suficiente, que se adapte a irregularidades del terreno para que realice lo siguiente:

Control mediante seguimiento por luz.

Percepción de obstáculos con a la finalidad de tomar una mejor ruta en base al obstáculo.

Encendido de luz nocturna automática.



Fuente: (Autores)

2.-Proyecto de identificación de colores mediante visión artificial, utilizando Matlab , y una web cam.



Fuente: (Autores)

BIBLIOGRAFÍA

Bibliografía básica

Barrientos Cruz, A. (2007). *Fundamentos de robótica*.

Kelly, R., & Santibáñez, V. (2003). Control de movimiento de robots manipuladores. Pearson educación.

Bibliografía complementaria

ArduinoModules. (2021). KY-021 Mini Magnetic Reed Switch Module. *ArduinoModulesInfo*.

<https://arduinomodulesinfo-1ba70b.ingress-haven.ewp.live/ky-021-mini-magnetic-reed-switch-module/>

Asimov, I. (2008). *El robot completo*. Madrid: Alamut/Bibliópolis.

https://es.wikipedia.org/w/index.php?title=Tres_leyes_de_la_rob%C3%B3tica&oldid=126886529

Barrientos, A. (2007). *Fundamentos de robótica*.

Barrientos Cruz—2007—Fundamentos de robótica.pdf. (s. f.).

Brazo robótico controlado por Arduino: Presupuesto. (s. f.). *Luis Llamas*. Recuperado 29 de junio de 2020, de

<https://www.luisllamas.es/brazo-robotico-arduino/>

Castaño, S. (2019a, marzo 18). *Motor paso a paso, Arduino*. Control Automático Educación.

<https://controlautomaticoeducacion.com/arduino/motor-paso-a-paso-arduino/>

Castaño, S. (2019b, diciembre 18). *Servomotor Arduino*. Control Automático Educación.

<https://controlautomaticoeducacion.com/arduino/servomotor-arduino/>

Dutour, G. (2019, septiembre 5). *Conexión y programación del OptoInterrupor LM393 o FC-03 | Robotica de la*

Costa. <https://roboticadelacosta.uy/tutorial-lm393-o-fc-03/>

Ebel, F., & Idler, S. (2019, diciembre 1). *FESTO fundamentos neumatica electroneumatica.pdf*.

Fraile, J. C. (s. f.). *Control y Programación de Robots*. Universidad de Valladolid Universidad de Valladolid E.T.S.

de Ingenieros Industriales. Recuperado 6 de agosto de 2020, de

https://alojamientos.uva.es/guia_docente/uploads/2013/452/42391/1/Documento6.pdf

González, V. (2002a). *Robots móviles*. Control automatizado y robótica 2002.

http://platea.pntic.mec.es/vgonzale/cyr_0204/cyr_01/robotica/movil.htm#plataformas

González, V. (2002b). *Transmisiones de movimiento en el robot*. Control automatizado y robótica 2002.

http://platea.pntic.mec.es/vgonzale/cyr_0204/cyr_01/robotica/sistema/transmisiones.htm

Isa, I. (s. f.). *Introducción a la robótica*. <http://isa.uniovi.es/~alonsog/Robotica/01%20Introduccion.pdf>

Legarreta, J., & Martínez, R. (s. f.). *Herramientas matemáticas para la localización espacial*.

Llamas, L. (2016, mayo 26). Controlar motores de corriente continua con Arduino y L298N. *Luis Llamas*.

<https://www.luisllamas.es/arduino-motor-corriente-continua-l298n/>

Los robots facilitan el trabajo de los humanos cada vez más. (2013, septiembre 13). *Robotnik*.

<https://www.robotnik.es/los-robots-facilitan-el-trabajo-de-los-humanos-cada-vez-mas/>

MakerSpace, Z. (s. f.). Curso de Arduino y robótica: Motor DC - Driver L298N. *Zaragoza MakerSpace*.

Recuperado 7 de agosto de 2020, de <https://zaragozmakerspace.com/index.php/lessons/curso-de-arduino-y-robotica-motor-dc-driver-l298n/>

Mitch Rc. (2014, junio 18). *F1 motores cc*. <https://www.slideshare.net/michellepamelarodriguezcorrales/f1-motores-cc-36046001/3%C3%A7>

mobile, hydraulictips. (2013, septiembre 24). *Pinzas hidráulicas*.

<https://www.mobilehydraulictips.com/hydraulic-grippers/>

Núñez, C. (2017). *Desarrollo de un sistema de control para el posicionamiento del servomecanismo de una cámara de identificación de vehículos infractores por semáforo mediante reconocimiento de imágenes utilizando visión artificial*.

PatagoniaTec. (s. f.). Moto Amarillo Para Arduino. *PatagoniaTec*. Recuperado 7 de agosto de 2020, de

<https://saber.patagoniatec.com/2014/06/tt-gear-yellow-motor-motorreductor-amarillo-para-arduino-arduino-argentina-ptec/>

Práctica 2 Cinemática de Robots. (s. f.).

Reyes, F., & Rodríguez, F. J. (2013). *MATLAB: Aplicado a la robótica y mecatrónica*. Alfaomega.

Robots de servicios seguridad. (s. f.). Recuperado 29 de junio de 2020, de

<https://www.google.com/search?q=Robots+de+servicios+seguridad&tbm=isch&ved=2ahUKEwjAu7fqgKjqAhVSWTABHURMDWYQ2->

cCegQIABAA&oq=Robots+de+servicios+seguridad&gs_lcp=CgNpbWcQA1CDkQJYz98CYJOCA2gAcAB4A4AB6ymIAeNhkgEHNi0xLjktM5gBAKABAaoBC2d3cy13aXotaW1n&sclient=img&ei=OWL6XoC7LtKywbkPxJi1sAY&bih=608&biw=1366&safe=active#imgrc=JcKvLT4eB4myeM

Salazar, R. (s. f.). *Introducción a la Robótica*. 20.

shoptronica. (s. f.). *Interruptor magnético Reed switch de potencia*. Recuperado 7 de agosto de 2020, de <https://www.shoptronica.com/interruptor-magnetico-pir-reed-switch/2387-interruptor-magnetico-reed-switch-de-potencia-8944748428929.html>

Sistemas de Locomoción de robots móviles. (s. f.). Recuperado 9 de diciembre de 2020, de http://www.esi2.us.es/~vivas/ayr2iaei/LOC_MOV.pdf

Valero, P. J. S. (s. f.). *Introducción a la Robótica Educativa Inteligente*. 80.

Xu, B., Shen, J., Liu, S., Su, Q., & Zhang, J. (2020). Research and Development of Electro-hydraulic Control Valves Oriented to Industry 4.0: A Review. *Chinese Journal of Mechanical Engineering*. <https://doi.org/10.1186/s10033-020-00446-2>

Yandún, A., & Sotomayor, N. (s. f.). *Planeación y seguimiento de trayectorias para un robot móvil*. 11.



Red de Investigación
Científica y Desarrollo
Tecnológico **Del Pacífico**


EDITORIAL
SAGA

