



Diseño e Implementación de una celda robotizada didáctica para el control de trayectorias en brazo robótico antropomórfico mediante programación offline con software educativo, para la aplicación al empaquetado de productos, en el laboratorio de Robótica Industrial de la Universidad de las Fuerzas Armadas Espe Sede Latacunga.

Dacto Yanza, Mayra Fernanda y Medina Medina, Ney Andrés

Departamento de Ciencias de la Energía y Mecánica

Carrera de Mecatrónica

Trabajo de Integración Curricular, previo a la obtención del título de Ingeniero en Mecatrónica

Ing. Singaña Amaguaña, Marco Adolfo

22 de agosto del 2023

Latacunga

Reporte de Verificación de Contenidos



Trabajo_titulación_Medina-Dacto.pdf

Scan details

Scan time:
August 22th, 2023 at 20:1 UTC

Total Pages:
104

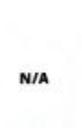
Total Words:
25823

Plagiarism Detection



Types of plagiarism		Words
Identical	1.3%	331
Minor Changes	0.4%	113
Paraphrased	2.9%	738
Omitted Words	0%	0

AI Content Detection



Text coverage
 AI text
 Human text

🔍 Plagiarism Results: (36)

🌐 Seguridad en robots colaborativos (COBOTS) - INTE... 0.6%
<https://www.integra-sti.com/noticias/seguridad-en-robots-co...>
 Saltar al contenido ...

🌐 ¿Qué es la Robótica Colaborativa? ▷ Aplicaciones y ... 0.4%
<https://www.edrobotics.com/blog/robotica-colaborativa-qu...>
 Inicio Sectores Productos Mezclado de caucho automático | Autobale Mixer Feeder ...

🌐 Microsoft Word - Andrade_Curay_TrabajodelIntegrac... 0.4%
<https://repositorio.espe.edu.ec/bitstream/21000/33784/1/t-e...>
 Administrador
 1 Carátula Repotenciación de la celda de soldadura robotizada que incluye el brazo robótico KUKA KR5-Arc, mesa posicionadora, sujetadore...

Ing. Singaña Amagaña Marco Adolfo
 C.C: 0501627889



Departamento de Ciencias de la Energía y Mecánica

Carrera de Mecatrónica

Certificación

Certifico que el trabajo de integración curricular: “Diseño e Implementación de una celda robotizada didáctica para el control de trayectorias en brazo robótico antropomórfico mediante programación offline con software educativo, para la aplicación al empaquetado de productos, en el laboratorio de Robótica Industrial de la Universidad de las Fuerzas Armadas Espe Sede Latacunga” fue realizado por los señores **Dacto Yanza, Mayra Fernanda y Medina Medina, Ney Andrés**, el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizada en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

Latacunga, 22 de agosto de 2023

Firma:

Ing. Singaña Amaguaña, Marco Adolfo

C.C: 0501627889



Departamento de Ciencias de la Energía y Mecánica

Carrera de Mecatrónica

Responsabilidad de Autoría

Nosotros, **Dacto Yanza, Mayra Fernanda** con cédula de ciudadanía N° 0605314350, y **Medina Medina, Ney Andrés** con cédula de ciudadanía N° 0605426188, declaramos que el contenido, ideas y criterios del trabajo de integración curricular: “**Diseño e Implementación de una celda robotizada didáctica para el control de trayectorias en brazo robótico antropomórfico mediante programación offline con software educativo, para la aplicación al empaquetado de productos, en el laboratorio de Robótica Industrial de la Universidad de las Fuerzas Armadas Espe Sede Latacunga**”, es de nuestra autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Latacunga, 22 de agosto de 2023

Firma:

Dacto Yanza, Mayra Fernanda

C.C.: 0605314350

Firma:

Medina Medina, Ney Andrés

C.C.: 0605426188



Departamento de Ciencias de la Energía y Mecánica

Carrera de Mecatrónica

Autorización de Publicación

Nosotros, **Dacto Yanza, Mayra Fernanda** con cédula de ciudadanía N° 0605314350, y **Medina Medina, Ney Andrés** con cédula de ciudadanía N° 0605426188, autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de integración curricular: “**Diseño e Implementación de una celda robotizada didáctica para el control de trayectorias en brazo robótico antropomórfico mediante programación offline con software educativo, para la aplicación al empaquetado de productos, en el laboratorio de Robótica Industrial de la Universidad de las Fuerzas Armadas Espe Sede Latacunga**”, en el Repositorio Institucional, cuyo contenido, ideas y criterios son de nuestra responsabilidad.

Latacunga, 22 de agosto de 2023

Firma:

Dacto Yanza, Mayra Fernanda

C.C.: 0605314350

Firma:

Medina Medina, Ney Andrés

C.C.: 0605426188

Dedicatoria

Empiezo dedicando este proyecto a Dios y la Virgen María, por las bendiciones recibidas en cada decisión tomada a nivel personal y académico.

A continuación, se lo dedico a mis queridos padres: Vinicio D. y Gladis Y. que, gracias a su sacrificio, consejos, cariño, regaños y su apoyo incondicional he logrado culminar con esta etapa en mi vida y son parte esencial en la realización de esta meta.

A mi hermana Jamilex, quien a pesar de la distancia ha sido mi motivación para no rendirme, que gracias a su compañía y respaldo me ha impulsado a seguir adelante y cumplir con mis sueños.

A mis abuelos: Ángel Dacto, Rosa Quinllin, Gerardo Yanza y Rosa Guananga por haberme premiado con los padres de quienes hoy gozo su compañía.

A mis tíos y primos, quienes con palabras de aliento y su preocupación, depositaron su confianza en cada decisión tomada; de una manera especial a quienes considero mis hermanos mayores: Rolando, Santiago, Bayron, Deysi, Greici y Ana; y con mucho cariño para quién estimo como mi segunda madre: Inés Dacto.

A mis primos: Edison, Katy, Joel, Shily, Yaja, y en especial para los más pequeños de la casa: Saúl, Javier, Matías, Bayron, Polette, Yuliet, Hanna, José, Stiven, Alison, Mateo y Johann; quienes se convirtieron en mi impulso para demostrar que uno debe ser mejor cada día.

Como no dedicárselo a los amigos y personas especiales que formaron parte de esta gran aventura: José B., Klever Ch., Alexander C., Mauricio A., Joel A., Jean C., F., Carlos L., Mishell I., Andrés V., Ney M., Tomás C., Edwin Ch., y en especial a los integrantes del Ballet Folklórico LLactacunga.

Mayra Fernanda Dacto Yanza

Dedicatoria

Este trabajo lo dedicó a mi abuela materna Luz Jimbo, sus consejos y enseñanzas fueron un pilar fundamental en mi desarrollo personal, a mi abuela paterna Josefa Barros que me brindó momentos felices e inolvidables, y por último a mi bisabuela Rosa Medina, que sin su apoyo este sueño no hubiera sido posible.

Ney Andrés Medina Medina

Agradecimiento

Empiezo agradeciendo a Dios y la Virgen María por las bendiciones recibidas en mi vida personal y académica; permitiéndome culminar con éxito una de mis muchas metas planteadas.

Agradezco a mis padres y hermana por ser quienes confiaron en mí a pesar de la distancia y los malos momentos; por todo su amor, comprensión y apoyo incondicional, les agradezco infinitamente de todo corazón.

Un agradecimiento a la vida por permitirme compartir este momento muy especial con mis abuelos, y cumplirles este sueño de superación.

Quiero agradecer de una manera muy especial a mi prima Deysi a quién considero una hermana, amiga y confidente; por haberme apoyado con consejos, palabras de ánimo y cariño en mis momentos de flaqueza.

Agradezco a mi familia en general por su preocupación y buenos deseos hacia mi persona, para la realización de mis metas.

A mi tutor de tesis Ing. Marco Singaña por su guía, colaboración y conocimientos contribuidos en la elaboración de este proyecto de titulación.

A mi compañero de tesis Ney Medina por su esfuerzo y dedicación para la culminación exitosa de este proyecto.

A mis amigos y esas personas especiales por el acolité, comprensión, salidas, cariño; ya que se convirtieron en la familia que escogí para la etapa de mi vida universitaria, y de una u otra manera contribuyeron en la materialización de este proyecto.

Mayra Fernanda Dacto Yanza

Agradecimiento

Quiero agradecer a Dios por permitirme culminar con éxito este sueño, a mi madre Norma Medina por la confianza, los consejos que siempre me han brindado y su apoyo incondicional en todo, mis logros académicos han sido posibles gracias a ella.

A mi padre Carlos Medina, a mi hermano Neyser Medina y a mi hermana Mercy Medina que han sido mi motivación para no rendirme y luchar por cumplir mis metas.

A mi tía Fanny Medina, a mi tío Manuel Alarcón y a mi primo Carlos Bedón, gracias a su apoyo incondicional y desinteresado pude avanzar y culminar la carrera.

A mi compañera de tesis Mayra Dacto que gracias a su esfuerzo hemos podido concluir este trabajo.

A mis amigos y amigas con los que compartí muchas vivencias que las llevo guardadas en mi corazón.

Ney Andrés Medina Medina

ÍNDICE DE CONTENIDOS

Carátula	1
Reporte de Verificación de Contenidos	2
Certificación	3
Responsabilidad de Autoría	4
Autorización de Publicación	5
Dedicatoria	6
Dedicatoria	7
Agradecimiento	8
Índice de contenidos	10
Índice de figuras	16
Índice de tablas	21
Resumen	23
Abstract	24
Capítulo I: Generalidades	25
Antecedentes	25
Planteamiento del problema	26
Descripción resumida del proyecto	28
Justificación, Importancia y Alcance	29
Objetivos	30
Objetivo General	30
Objetivo Específicos	30

Metodología	31
<i>Metodología documental bibliográfico</i>	31
<i>Metodología experimental</i>	31
<i>Metodología deductiva</i>	32
<i>Metodología Inductiva</i>	32
Hipótesis	32
<i>Variable independiente</i>	32
<i>Variabes dependientes</i>	32
Capítulo II: Fundamentación e Investigación Teórica	33
Introducción	33
Robots Industriales	33
<i>Robótica</i>	33
<i>Robótica Industrial</i>	34
<i>Robótica Colaborativa</i>	49
Celda Robotizada	58
<i>Definición</i>	58
<i>Etapas de diseño de una celda robotizada</i>	58
<i>Modos de funcionamiento</i>	58
<i>Layout</i>	59
<i>Distribución de los equipos en la celda robotizada</i>	60
<i>Disposición del robot en la celda robotizada</i>	60

<i>Características del sistema de control de la celda robotizada</i>	61
<i>Modos operativos de los cobots en una celda robotizada</i>	62
Seguridades para el trabajo en una celda robotizada	64
<i>Elementos de Seguridad de una celda robotizada</i>	64
<i>Normativa Legal para el Trabajo en las Celdas Robotizadas con COBOTS</i>	65
<i>Seguridad en robots colaborativos</i>	66
Software de programación de robots industriales.	68
<i>Lenguajes de programación de robots</i>	68
<i>Métodos de programación</i>	71
<i>Programación en Línea</i>	72
<i>Programación fuera de línea</i>	73
<i>Métodos actuales de programación de cobots</i>	75
<i>Software para programación de cobots.</i>	76
Capítulo III: Selección de componentes	81
Introducción	81
Planificación del diseño	81
<i>Definición de las necesidades</i>	81
<i>Definición de las especificaciones técnicas</i>	82
<i>Matriz QFD</i>	83
Selección de robot antropomórfico.	85
<i>Características de selección.</i>	86

<i>Opción 1: Universal Robots - UR3e</i>	86
<i>Opción 2: FANUC CR-4iA</i>	87
<i>Opción 3: UFACTORY Lite 6</i>	88
<i>Selección del cobot</i>	90
<i>Efecto final</i>	91
<i>Vacuum Gripper Lite 6</i>	91
Selección de software de programación offline.....	92
<i>Opción1. Simulador Robótico GAZEBO</i>	92
<i>Opción 2. Simulador SIMUMATIK</i>	93
<i>Opción 3. Simulador RoboDK</i>	94
Capítulo IV: Diseño y Construcción de la Celda	97
Parámetros de diseño.....	97
Diseño de la celda robotizada	98
<i>Elementos activos</i>	98
<i>Elementos pasivos</i>	98
<i>Disposición de la Celda Robotizada</i>	99
<i>Sistema de control</i>	100
<i>Seguridades de la celda</i>	101
Diseño de la Mesa	102
<i>Análisis Estructural</i>	104
Diseño de la banda transportadora	108

<i>Parámetros de diseño</i>	108
<i>Circuito Electrónico</i>	110
Construcción de la celda robotizada	110
<i>Construcción de la estructura de la celda</i>	110
<i>Instalación de accesorios en la mesa de trabajo</i>	111
Comunicación entre el robot y dispositivos.....	113
<i>Comunicación PC – COBOT</i>	113
<i>Comunicación PC - Banda</i>	116
<i>Comunicación COBOT - Banda</i>	116
Programación offline en el software seleccionado	117
Capítulo V: Pruebas y Resultados	136
Parámetros de funcionamiento de la aplicación de empaquetado del proyecto	136
Operación del brazo robótico y ventosa	136
Operación de la banda transportadora	137
Prueba de comunicación entre el robot y dispositivos.....	138
Prueba de la aplicación completa	139
<i>Prueba de posición (XYZ)</i>	139
<i>Prueba de posición (X, Y, Z) con las trayectorias realizadas</i>	141
<i>Pruebas de trayectorias</i>	143
<i>Pruebas de velocidad y repetibilidad</i>	150
<i>Validación didáctica de la celda robotizada</i>	155

<i>Validación del control de trayectorias en el brazo robótico antropomórfico</i>	161
<i>Planteamiento de Hipótesis</i>	162
Capítulo VI: Conclusiones y Recomendaciones	166
Conclusiones	166
Recomendaciones	168
Bibliografía	170
Anexos	177

ÍNDICE DE FIGURAS

Figura 1 <i>Partes de un brazo robótico industrial.</i>	37
Figura 2 <i>Gráfico de tasa de crecimiento anual en instalaciones de robots.</i>	39
Figura 3 <i>Partes constitutivas de un brazo robótico.</i>	40
Figura 4 <i>Partes constitutivas de un robot industrial</i>	40
Figura 5 <i>Partes constitutivas de un manipulador.</i>	41
Figura 6 <i>Tipos de articulaciones.</i>	42
Figura 7 <i>Orientación de los eslabones.</i>	42
Figura 8 <i>Movimientos de la muñeca.</i>	43
Figura 9 <i>TCP de un brazo robótico.</i>	44
Figura 10 <i>Dispositivos de entrada y salida de un robot industrial.</i>	45
Figura 11 <i>Características de un robot tradicional y un COBOT.</i>	51
Figura 12 <i>Partes constitutivas del COBOT UFACTORY Lite6.</i>	56
Figura 13 <i>Características técnicas de robots colaborativos de UFACTORY.</i>	56
Figura 14 <i>Modos de interacción entre cobots y los humanos.</i>	63
Figura 15 <i>Interfaz de la plataforma de emulación SIMUMATIK.</i>	78
Figura 16 <i>Interfaz de la plataforma de emulación RoboDK.</i>	79
Figura 17 <i>Interfaz de la plataforma de emulación GAZEBO.</i>	80
Figura 18 <i>Cobot de la empresa Universal Robots modelo UR3e.</i>	87
Figura 19 <i>Cobot de la empresa FANUC modelo CR-4iA.</i>	88
Figura 20 <i>Cobot de la empresa UFACTORY modelo Lite 6.</i>	89
Figura 21 <i>Vacuum Gripper Lite 6 de la empresa UFACTORY.</i>	92
Figura 22 <i>Área de trabajo del COBOT Lite 6.</i>	99
Figura 23 <i>Distribución centralizada de la celda.</i>	100
Figura 24 <i>Sistema descentralizado de control</i>	101

Figura 25 <i>Botón paro de emergencia</i>	101
Figura 26 <i>Señalética de seguridad</i>	102
Figura 27 <i>Diseño de la estructura de la mesa</i>	102
Figura 28 <i>Diseño del tablero de la mesa</i>	103
Figura 29 <i>Ensamble de la estructura con el tablero de la mesa</i>	103
Figura 30 <i>Distribución de la carga y restricciones en la estructura de la mesa.</i>	105
Figura 31 <i>Resultados del análisis estático en la estructura de la mesa para carga vertical....</i>	106
Figura 32 <i>Distribución y restricciones en la estructura de la mesa para carga horizontal.</i>	107
Figura 33 <i>Análisis estático en la estructura de la mesa para carga horizontal.</i>	108
Figura 34 <i>Diseño CAD de la banda transportadora.</i>	109
Figura 35 <i>Esquemático del circuito electrónico</i>	110
Figura 36 <i>Diagrama de proceso de la construcción estructural de la celda</i>	111
Figura 37 <i>Diagrama de proceso instalación de accesorios en la celda</i>	112
Figura 38 <i>Esquema de conexión del COBOT LITE 6.</i>	112
Figura 39 <i>Construcción de la celda robotizada</i>	113
Figura 40 <i>Diagrama de proceso para configuración IP del robot y ordenador.</i>	114
Figura 41 <i>Diagrama de conexión de una carga resistiva al controlador</i>	115
Figura 42 <i>Diagrama de flujo del funcionamiento de la banda.</i>	116
Figura 43 <i>Diagrama de flujo del funcionamiento de la banda.</i>	117
Figura 44 <i>Ejecución del Layout de la celda robotizada.</i>	117
Figura 45 <i>Creación de Frame y ajuste de posición</i>	118
Figura 46 <i>Creación y configuración de los puntos de la trayectoria.</i>	119
Figura 47 <i>Configuración de nombre del punto generado.</i>	119
Figura 48 <i>Creación del programa para generar la trayectoria.</i>	120
Figura 49 <i>Creación de movimientos necesarios para completar una trayectoria.</i>	120
Figura 50 <i>Secuencia de instrucciones para crear una "Instrucción de evento".</i>	121

Figura 51 <i>Generación del evento de encendido y succión de la ventosa.</i>	122
Figura 52 <i>Creación de la instrucción de velocidad y configuración de parámetros.</i>	122
Figura 53 <i>Creación de la instrucción de Llamado a Programa.</i>	123
Figura 54 <i>Entrada digital del COBOT y código de activación de la ventosa.</i>	123
Figura 55 <i>Definición de movimientos e instrucciones para la generación de trayectorias.</i>	124
Figura 56 <i>Condiciones necesarias para la creación de la trayectoria.</i>	124
Figura 57 <i>Instrucciones del programa PINICPRO.</i>	125
Figura 58 <i>Instrucciones del programa TSPH.</i>	125
Figura 59 <i>Programación del programa Alarma_ON.</i>	125
Figura 60 <i>Codificación del programa Alarma_OFF.</i>	126
Figura 61 <i>Instrucciones del programa IOST1.</i>	126
Figura 62 <i>Codificación del programa IOSP.</i>	127
Figura 63 <i>Programación del programa PF.</i>	127
Figura 64 <i>Collage de imágenes de la ejecución de la trayectoria.</i>	128
Figura 65 <i>Collage de imágenes de la segunda trayectoria.</i>	129
Figura 66 <i>Selección de la opción Post-procesador.</i>	129
Figura 67 <i>Selección del post-procesador denominado Lite6.</i>	130
Figura 68 <i>Selección de la opción "Generar programa del robot".</i>	130
Figura 69 <i>Código en lenguaje Python post procesado desde RoboDK.</i>	131
Figura 70 <i>Archivo .py</i>	131
Figura 71 <i>Procedimiento para obtener el archivo .py</i>	131
Figura 72 <i>Creación del archivo con extensión ".tar".</i>	132
Figura 73 <i>Selección de la extensión .gzip para generar un archivo .gz</i>	133
Figura 74 <i>Archivo T1.tar.gz generado.</i>	133
Figura 75 <i>Selección de carpeta y opción de importación.</i>	133
Figura 76 <i>Selección del archivo a cargarse en el controlador del robot.</i>	134

Figura 77 <i>Carga del archivo T1.tar.gz, ejecución del mismo y aceptación de mensajes.</i>	134
Figura 78 <i>Ejecución del programa de la trayectoria creada en RoboDK.</i>	135
Figura 79 <i>Área de trabajo para la aplicación de empaquetado.</i>	136
Figura 80 <i>Juego de ventosas del brazo robótico.</i>	137
Figura 81 <i>Operación del brazo robótico con una carga de 113 g.</i>	137
Figura 82 <i>Funcionamiento de la banda.</i>	138
Figura 83 <i>Prueba de la comunicación del robot y los dispositivos.</i>	139
Figura 84 <i>Posiciones configuradas en el layout de RoboDK.</i>	139
Figura 85 <i>Posiciones compensadas.</i>	142
Figura 86 <i>Movimiento articular en RoboDK.</i>	144
Figura 87 <i>Movimiento articular ejecutado en la celda.</i>	144
Figura 88 <i>Movimiento lineal en RoboDK.</i>	145
Figura 89 <i>Movimiento lineal ejecutado en la celda.</i>	145
Figura 90 <i>Movimiento Circular</i>	148
Figura 91 <i>Movimiento circular ejecutado en la celda.</i>	148
Figura 92 <i>Trayectorias para la aplicación de empaquetado.</i>	155
Figura 93 <i>Punto inicial de la trayectoria sometida a pruebas de velocidad y repetibilidad.</i>	151
Figura 94 <i>Punto final de la trayectoria sometida a pruebas de velocidad y repetibilidad.</i>	151
Figura 95 <i>Parámetros de velocidad configurada en RoboDK.</i>	151
Figura 96 <i>Resultados de la pregunta 1.</i>	156
Figura 97 <i>Resultados de la pregunta 2.</i>	157
Figura 98 <i>Resultados de la pregunta 3.</i>	157
Figura 99 <i>Resultados de la pregunta 4.</i>	158
Figura 100 <i>Resultados de la pregunta 5.</i>	158
Figura 101 <i>Resultados de la pregunta 6.</i>	159
Figura 102 <i>Resultados de la pregunta 7.</i>	159

Figura 103 <i>Resultados de la pregunta 8.</i>	160
Figura 104 <i>Resultados de la pregunta 9.</i>	160
Figura 105 <i>Resultados de la pregunta 10.</i>	161
Figura 106 <i>Selección del valor de nivel de confianza en función del error.</i>	165

ÍNDICE DE TABLAS

Tabla 1	<i>Empresas fabricantes de robots colaborativos.</i>	54
Tabla 2	<i>Características técnicas de robots colaborativos de la empresa Universal Robots.</i>	57
Tabla 3	<i>Lenguajes de programación de robots industriales.</i>	69
Tabla 4	<i>Principales desarrolladores de programas para la programación fuera de línea.</i>	75
Tabla 5	<i>Necesidades en función de los usuarios.</i>	81
Tabla 6	<i>Especificaciones técnicas de la celda robotizada.</i>	82
Tabla 7	<i>Escala de correlaciones.</i>	82
Tabla 8	<i>Matriz QFD del proyecto (Quality Function Deployment).</i>	83
Tabla 9	<i>Resultados de la matriz QFD.</i>	84
Tabla 10	<i>Sistemas y subsistemas del proyecto.</i>	85
Tabla 11	<i>Características deseables para la selección del robot.</i>	86
Tabla 12	<i>Características del cobot Universal Robots - UR3e.</i>	86
Tabla 13	<i>Características del cobot FANUC CR-4iA.</i>	88
Tabla 14	<i>Características del cobot UFACTORY Lite 6.</i>	89
Tabla 15	<i>Selección del brazo robótico colaborativo para la celda robotizada.</i>	90
Tabla 16	<i>Características del Vacuum Gripper Lite 6.</i>	91
Tabla 17	<i>Características del software de simulación GAZEBO.</i>	92
Tabla 18	<i>Características del software de simulación SIMUMATIK.</i>	93
Tabla 19	<i>Características del software de simulación RoboDK.</i>	94
Tabla 20	<i>Parámetros de selección de software de programación offline.</i>	95
Tabla 21	<i>Elementos activos que conforman la celda robotizada.</i>	98
Tabla 22	<i>Elementos pasivos que conforman la celda robotizada.</i>	99
Tabla 23	<i>Propiedades mecánicas del acero ASTM A36.</i>	104
Tabla 24	<i>Análisis estático de la estructura de la mesa para carga vertical.</i>	105

Tabla 25	<i>Resultados del análisis estático en la estructura de la mesa para carga horizontal.</i>	107
Tabla 26	<i>Parámetros de funcionamiento.</i>	115
Tabla 27	<i>Comparación de posiciones entre RoboDK y el controlador del robot.</i>	140
Tabla 28	<i>Comparación de valores compensados y del controlador.</i>	142
Tabla 29	<i>Datos de desplazamiento en Y.</i>	145
Tabla 30	<i>Comparación entre punto inicial y final de las coordenadas X y Z.</i>	147
Tabla 31	<i>Valores de radio y diámetros simulados y reales.</i>	149
Tabla 32	<i>Datos para pruebas de velocidad.</i>	152
Tabla 33	<i>Datos para pruebas de velocidad.</i>	153
Tabla 34	<i>Pruebas realizadas.</i>	162
Tabla 35	<i>Cálculo del método estadístico.</i>	164

Resumen

Este proyecto tiene como objetivo el control de trayectorias necesarias para la aplicación de empaquetado de productos, utilizando un entorno virtual con la metodología de programación fuera de línea desarrollado en el software RoboDK. Luego, se importarán dichas trayectorias al controlador del cobot UFACTORY Lite 6 para evaluar la precisión de los movimientos. Se verificará la congruencia entre las coordenadas de posición simuladas y las medidas reales. El proyecto se inicia con el análisis de modelos recientes de robots antropomórficos y sus aplicaciones. Luego, se divide en tres fases clave: modelado de la celda robotizada, creación de un entorno virtual para la programación fuera de línea y configuración de la comunicación entre los componentes. La primera fase incluye la representación en 3D de la celda y sus partes mediante un software CAD. La segunda etapa se enfoca en establecer un entorno virtual en el software RoboDK. En la última fase, se conectan los elementos para realizar pruebas y verificar el funcionamiento. El proyecto culmina en la interacción de los estudiantes con una práctica desarrollada en la celda robotizada, trabajando con la metodología de programación fuera de línea, para validar su aplicabilidad educativa.

Palabras clave: programación fuera de línea, entorno virtual, RoboDK, UFACTORY Lite 6, control de trayectorias.

Abstract

The objective of this project is to control the trajectories required for the product packaging application, using a virtual environment with the offline programming methodology developed in RoboDK software. Then, these trajectories will be imported to the UFACTORY Lite 6 cobot controller to evaluate the accuracy of the movements. The congruence between the simulated position coordinates and the real measurements will be verified. The project starts with the analysis of recent anthropomorphic robot models and their applications. It is then divided into three key phases: modeling of the robotic cell, creation of a virtual environment for offline programming, and configuration of communication between components. The first phase includes the 3D representation of the cell and its parts using CAD software. The second stage focuses on setting up a virtual environment in RoboDK software. In the last phase, the elements are connected for testing and verification of operation. The project culminates in the interaction of students with a practice developed in the robotic cell, working with the offline programming methodology, to validate its educational applicability.

Keywords: offline programming, virtual environment, RoboDK, UFACTORY Lite 6, path control.

Capítulo I

Generalidades

Antecedentes

La evolución de la robótica colaborativa en los procesos industriales tiene un impacto incremental en las líneas de manufactura, disponiendo la mínima intervención del operario en un proceso de producción, cuidando la integridad física de éste y así la entereza de la línea de producción (EDS Robotics, 2022).

La programación off-line se refiere a programar robots fuera del entorno de producción, de esta manera se elimina el tiempo de inactividad de la manufactura causado por la programación on-line de dichos elementos.

La simulación y la programación fuera de línea, permiten estudiar múltiples escenarios de una celda robotizada antes de configurar dicha celda de producción. Los errores que se cometen comúnmente al diseñar una celda de trabajo se pueden predecir y corregir antes de su implementación.

La metodología de programación fuera de línea ofrece la posibilidad de programar robots virtualmente en 3D, desde un ordenador, utilizando un software que permita realizar este tipo de programación, esta herramienta puede ser utilizada en diferentes situaciones: si su finalidad es la optimización de procesos actuales o la correcta implementación de futuros procesos (Aljama, 2011).

Todo esto ocurre sin necesidad de interrumpir la producción, con el mínimo riesgo y sin pérdidas de productividad por tiempos en instalación y programación en la misma fábrica.

Esta metodología requiere altos costes iniciales debido a la formación del personal, adquisición de las licencias de software que permita simular los movimientos del robot, además de la virtualización de los elementos que constituyen la celda robotizada, sin embargo, esta

inversión ha valido la pena, ya que los plazos de entrega de los nuevos programas pueden reducirse de semanas a días, lo que permite automatizar la producción en poco tiempo (Albea Sánchez, 2004b).

Planteamiento del problema

Un robot se considera programado en línea cuando se utiliza durante la etapa de desarrollo del programa. Este enfoque representa el método convencional de llevar a cabo la programación, lo que implica tener acceso físico al robot durante todo el proceso de desarrollo. No obstante, esta modalidad requiere no solo la posesión física del robot, sino también la programación precisa de las trayectorias en el entorno donde se llevará a cabo la tarea posteriormente, a fin de garantizar una programación adecuada.

Una desventaja de la programación en línea es no poder prevenir fallos y posibles daños en los mecanismos del brazo robótico y en el entorno de producción, ya que, los programas no pueden verificarse o validarse antes de su implementación, lo que representa un riesgo para los usuarios puesto que permanecen cerca del área de operación del robot.

La creciente competencia en la industria y el aumento en la demanda han llevado a una mayor necesidad de aumentar la productividad. Esta situación ha tenido un impacto significativo en el ámbito de los robots industriales, que ahora están explorando nuevas formas de programación.

La programación fuera de línea emerge como una solución que puede potenciar el rendimiento de los robots industriales. Como resultado, se puede lograr una mayor productividad y reducir los períodos de inactividad durante la fase de desarrollo de la producción. Mediante el uso de simulaciones, es posible analizar la evolución de los procesos de producción, optimizarlos y reducir los tiempos de fabricación de un producto. Cada vez más

se tiende a utilizar la programación off-line, lo cual permite desarrollar programas sin la necesidad de disponer del robot.

El problema motivo del presente trabajo radica en la potenciación o incrementar el equipamiento del laboratorio de Robótica Industrial de la Universidad de las Fuerzas Armadas ESPE Sede Latacunga, para la creación de algoritmos de control mediante programación fuera de línea en el laboratorio, ya que, hasta ahora el proceso de control de los brazos robóticos existentes en el laboratorio antes mencionado, ha sido únicamente por programación on-line, es decir mediante un panel donde se programa cada una de las posiciones de toda la trayectoria que se desee generar, o desarrollar la trayectoria en base al posicionamiento punto a punto del manipulador.

Con la implementación de la celda robotizada didáctica integrada con software de programación off-line y un nuevo modelo de robot antropomórfico de entrenamiento, se da solución a la necesidad existente en el laboratorio, pues permite que el usuario pueda familiarizarse y aprender cómo se generan las trayectorias de movimiento que realiza el robot.

Además, con el proyecto se realizarán pruebas con estudiantes que tengan conocimientos afines a la robótica para verificar los beneficios de la programación fuera de línea.

También, en la celda de trabajo que se implementará, se controlan las trayectorias que realiza el robot en una aplicación de empaquetado de productos con esta metodología de programación, lo cual brindará a los usuarios la experiencia, instrucción y entendimiento acerca de la creación de los programas de procesos industriales, utilizando un modelo 3D de toda la celda de producción, en donde, se visualiza una simulación de la línea de manufactura, que permite optimizar las operaciones que se están realizando.

Descripción resumida del proyecto

Este proyecto tiene como objetivo programar las trayectorias requeridas para un proceso industrial específico (empaquetado de productos) en modo fuera de línea, mediante un entorno virtual utilizando un modelo 3D, para su posterior implementación en el controlador del brazo robótico, de forma que se pueda validar que las trayectorias generadas en el software son las mismas que el brazo robótico ha realizado, verificando su precisión y exactitud en los movimientos; para lo cual se corroborará las coordenadas de la posición final que resulta en la simulación, en comparación con las coordenadas físicas medidas, tras finalizar la trayectoria previamente descrita.

El proyecto parte del análisis de nuevos modelos de robots antropomórficos y sus aplicaciones. En base a estos dos análisis se realizará la implementación de una celda robotizada.

El diseño e implementación se desarrollará en 3 etapas principales: modelamiento de la celda robotizada con sus elementos activos y pasivos, desarrollo del entorno virtual para la generación de trayectorias mediante un software de programación off-line y la comunicación del brazo robótico con los diferentes dispositivos.

La primera etapa consiste en el modelamiento de la celda robotizada mediante un software CAD, en el cual se pueda visualizar de manera cercana a la realidad los elementos activos y pasivos que conformarán la misma, además de la ubicación correspondiente para el brazo robótico.

La segunda etapa consiste en el desarrollo del entorno virtual de la celda de trabajo en el software de programación offline. Existen diversos tipos de software que permiten la realización de esta etapa, en este caso el software seleccionado será el más idóneo y apto en función de los criterios de trabajos requeridos.

Como última etapa se tiene la comunicación de todos los elementos de tal forma que se puedan realizar pruebas y verificaciones de funcionamiento de todos los accesorios pertenecientes a la celda robotizada; en esta fase del proyecto como tal, se realizarán participaciones interactivas de la celda, el robot, la metodología de programación offline, con operarios que tengan previo conocimiento de conceptos básicos de robótica, confirmando así el nivel de manipulación didáctico que presenta la celda construida.

Justificación, Importancia y Alcance

El tema propuesto se justifica debido al desarrollo tecnológico, mismo que, en las instituciones educativas juega un papel importante, ya que sus usuarios adquieren nuevos conocimientos y a su vez tienen la oportunidad de manipular las nuevas tecnologías, lo que les ayudará en su formación académica. Los nuevos conocimientos adquiridos en el laboratorio se pueden aplicar en la vida laboral.

En el proceso de control de trayectorias, se aprovechará la programación fuera de línea, realizado por medio de un software, mismo que permitirá disminuir el tiempo de inactividad de producción comparado con la programación en línea.

En el año 2004, Albea Sánchez en un estudio realizado ha comentado varias ventajas, mencionando que:

La programación off-line en un brazo robótico (COBOT), define la importancia del proyecto propuesto, ya que permitirá familiarizar a los distintos usuarios con procesos industriales y manipular nuevos equipos con tecnología de última generación, así como el uso de una nueva metodología de programación, que es muy utilizado en empresas con procesos robotizados a nivel industrial; teniendo como principal objetivo el salvaguardar la integridad del usuario; a más de reducir los períodos de inactividad producidos por el paro de la maquinaria en el proceso de producción (p. 20).

El alcance de la temática propuesta incluye la construcción e implementación de la celda robotizada didáctica, misma que contará con un brazo robótico antropomórfico (COBOT), que se programará bajo la modalidad offline en el control de las trayectorias que éste realice; para complementar lo anteriormente mencionado se enfoca el desarrollo de una aplicación de empaquetado de productos, haciendo referencia a los procesos realizados en los sectores de producción industrial.

Objetivos

Objetivo General

- Diseñar e implementar una celda robotizada didáctica para el control de trayectorias en un brazo robótico antropomórfico mediante programación offline con software educativo, para la aplicación al empaquetado de productos, en el Laboratorio de Robótica Industrial de la Universidad de las Fuerzas Armadas ESPE Sede Latacunga.

Objetivo Específicos

- Investigar las necesidades del laboratorio de robótica industrial y los sistemas robóticos didácticos existentes.
- Realizar el análisis del estado del arte en relación a nuevos modelos de robots antropomórficos y sus aplicaciones.
- Contrastar los programas compatibles con el brazo robótico seleccionado para programación offline, en función de la existencia de librerías necesarias para el funcionamiento del robot, y su posterior adquisición.
- Diseñar y construir la estructura mecánica de la celda robotizada, sus dimensiones y disposición de los elementos para su aplicación.
- Diseñar la estructura y control de la banda transportadora en función del producto seleccionado para la aplicación de empaquetado.

- Ensamblar el brazo robótico adquirido con la estructura mecánica para la obtención de la celda robotizada planteada.
- Configurar el sistema de comunicaciones entre los accesorios de la celda y el robot colaborativo.
- Programar utilizando la metodología off-line para el control de trayectorias en el brazo robótico, enfocado en el desarrollo de empaquetado de productos, en el software seleccionado con anterioridad.
- Exportar el código de programación off-line al controlador del brazo robótico seleccionado y realizar pruebas parciales de configuración y funcionalidad de dispositivos.
- Validar la hipótesis mediante pruebas de funcionalidad total de la aplicación.

Metodología

Para el avance del proyecto centrado en el diseño e implementación de una celda robotizada educativa en la Universidad de las Fuerzas Armadas ESPE - Sede Latacunga, se emplearán las siguientes metodologías:

Metodología documental bibliográfico

El procedimiento mencionado se llevará a cabo a través de la investigación y el análisis exhaustivo de diversas fuentes bibliográficas pedagógicas. Estas fuentes estarán relacionadas con celdas robotizadas educativas que incluyan brazos robóticos tipo COBOT. También se abordarán temas que sean pertinentes para la ejecución del proyecto de integración curricular.

Metodología experimental

Este enfoque se llevará a cabo fundamentándose en las funciones principales del brazo robótico colaborativo, buscando optimizar la programación de sus movimientos a través de un software de programación fuera de línea. Mediante la realización de pruebas exhaustivas, se

examinará minuciosamente el comportamiento de la aplicación de empaquetado generada. Los datos recopilados y las conclusiones derivadas de este análisis serán de vital importancia para validar la eficacia y adecuado funcionamiento del proceso de la programación offline para el desarrollo de procesos industriales.

Metodología deductiva

Por medio de la ejecución de esta metodología se logrará conseguir soluciones singulares en el progreso del proyecto, basadas en principios y conocimientos generales relacionados al control de trayectorias del robot seleccionado, mediante programación offline con el software educativo; inquiriendo esencialmente en investigaciones semejantes al control de trayectorias con programación offline en brazos antropomórficos.

Metodología Inductiva

Con la ejecución de esta metodología se conseguirán conclusiones generales, basadas en datos y conocimientos, tras el cumplimiento de objetivos específicos desarrollados durante el control de trayectorias con programación offline, para el proceso de empaquetado de productos.

Hipótesis

¿El diseño e implementación de una celda robotizada didáctica integrada con software de programación off-line permitirá controlar las trayectorias del brazo robótico antropomórfico, en una aplicación de empaquetado de productos?

Variable independiente

Celda robotizada integrada con software de programación offline.

Variables dependientes

Controlar las trayectorias del brazo robótico antropomórfico.

Validación didáctica de la celda robotizada.

Capítulo II

Fundamentación e Investigación Teórica

Introducción

En este capítulo se describen los conceptos utilizados para el desarrollo de la celda robotizada didáctica para el control de trayectorias en un brazo robótico antropomórfico mediante programación offline con software educativo.

Las celdas robotizadas son espacios de trabajo versátiles y flexibles, puesto que permiten la optimización de los tiempos de manufactura, además de aumento sustancial de la productividad y de la repetibilidad de la calidad del producto.

Por otro lado, la programación de brazos robóticos es esencial para el correcto funcionamiento de las celdas robotizadas, se busca constantemente reducir el tiempo requerido para la programación de dichas estaciones, esto ha determinado que se busquen nuevas alternativas de programación de robots industriales.

Robots Industriales

Robótica

Explorando diversas perspectivas del concepto de robótica, según lo planteado por Reyes (2011), se establece que "la robótica constituye un campo científico que se ocupa de la investigación y evolución de una categoría específica de sistemas mecánicos conocidos como robots manipuladores. Éstos están concebidos para llevar a cabo una amplia gama de tareas en ámbitos industriales, científicos, domésticos y comerciales" (p. 3). Este enfoque contribuye a una comprensión panorámica del significado intrínseco de la robótica.

La robótica es una ciencia interdisciplinaria que se deriva de áreas como la ingeniería mecánica, eléctrica, electrónica, biomédica y ciencias de la computación. Su objetivo es estudiar, diseñar, desarrollar, programar, producir y aplicar robots. Según la RAE (Real

Academia Española) un robot se define como una máquina o dispositivo electrónico programable con la capacidad de manipular objetos y llevar a cabo una variedad de operaciones. Esto implica la habilidad de procesar información y tomar acciones en concordancia con dicha información.

En el año 2022, EmpresaActual argumentó que la robótica se puede dividir en tres sectores, que se describen a continuación:

- *Robótica de Servicio:* Son creados con la finalidad de llevar a cabo tareas en beneficio de las personas, y están diseñados con propósitos específicos de prestación de servicios. Ejemplos de éstos incluyen robots de cocina, aspiradoras automatizadas y robots limpiadores de piscinas, a menudo denominados coloquialmente como "pulpos".
- *Robótica Social:* Se orienta hacia la creación de robots concebidos para interactuar con las personas en entornos sociales. Estos robots operan siguiendo pautas y normas sociales adaptadas a su propósito. Ejemplos de robótica social abarcan desde aplicaciones de asistentes de voz hasta robots educativos.
- *Robótica Industrial:* Es ampliamente utilizada en variados sectores industriales, gracias a su aptitud para proporcionar celeridad, exactitud y eficiencia en labores, sobre todo en aquellas que entrañan riesgos o son monótonas para los trabajadores (párr. 5-7).

Robótica Industrial

Tomando como referencia la publicación de Empresa Actual y con el objetivo de comprender la evolución del desarrollo de los robots industriales, se expone lo siguiente:

Este tipo de robótica nace en 1937 en Inglaterra, tras la creación de la primera grúa automatizada por Bill Taylor; tiempo después los robots fueron enfocados para realizar ciertas tareas lógicas y con movimientos cíclicos.

En la segunda mitad del siglo XX, tras el término de la Segunda Guerra Mundial, se marcó el inicio de la utilización de los primeros robots de grandes dimensiones. Durante la década de los años 60, la empresa pionera en robótica, Unimation Inc., logró crear el primer robot programable enfocado en la transferencia de objetos.

Los avances continuaron en las décadas de los 60 y 70, dando lugar a la aparición de brazos mecánicos más sofisticados. Sin embargo, fue en los años 80 cuando esta tecnología experimentó su fase de máximo desarrollo, siendo reconocida como el punto de partida de la Era Robótica. Durante este periodo, la producción y comercialización de robots incrementó en un 80%. Los primeros robots industriales surgieron en los Estados Unidos, los años 80 y 90 también fueron testigos del avance de esta tecnología en diversas naciones de Europa y Asia, con especial énfasis en países como Suecia y Japón (párr. 14-16).

En el siglo XXI se desarrolla la revolución industrial o mejor conocida como Industria 4.0, misma se enfoca en la producción y operación de distintos procesos haciendo uso de técnicas innovadoras y tecnologías inteligentes; es por ello que se considera a la robótica industrial como un pilar esencial en la Industria 4.0, ya que se permite el incremento de productividad, disminución de errores en cadenas de producción y optimización de recursos en los diferentes procesos industriales.

Este campo perteneciente a la ingeniería es orientado a la definición, diseño, desarrollo y fabricación de los distintos robots industriales, permitiendo la automatización de las tareas realizadas por el ser humano; haciendo estas labores de manera más rápida, precisa, consecutiva, sin pausas, pero sobre todo salvaguardando la integridad física del operario; permitiendo a una empresa incrementar los índices de eficiencia, calidad y tiempos de producción en ciclos repetitivos y peligrosos.

En el ámbito de la robótica industrial, varios autores han concordado en la definición que establece que "la robótica industrial se caracteriza por ser un sistema de control automatizado, reprogramable, que opera en tres o más ejes" (Sicma21, 2021, p. 1), lo que ha contribuido al entendimiento general acerca de esta disciplina.

Continuando con la explicación realizada en EmpresaActual (2022), se indica que:

Esta categoría de robótica está enfocada en la aplicación de robots en entornos de fabricación y producción, especialmente en la industria automotriz y manufacturera. Entre los tipos más reconocidos y utilizados se encuentran los brazos mecánicos, que se utilizan para la manipulación, ajuste y movimiento de diversos materiales, piezas y herramientas. En la actualidad, se ha presenciado la incorporación de elementos de inteligencia artificial (IA) y la integración con el Internet de las Cosas (IoT) en estos autómatas. Estos avances permiten llevar a cabo tareas altamente complejas con capacidades de aprendizaje, lo que resulta en una ejecución más rápida y eficiente, reduciendo errores en el proceso.

Conforme a la Federación Internacional de Robótica, la robótica industrial afrontará nuevos desafíos, como:

- Diferenciar entre una máquina y una persona al responder con mayor precisión a su entorno circundante.
- Mayor flexibilidad en los procesos industriales, ejecutando control de calidad en los productos finales, así como aumentando la eficacia en una cadena de suministro.
- Facilidad de programación para que las PYMES tengan acceso a este servicio; y a su vez apoyar el crecimiento del modelo de negocio *robots as a service*, donde empresas se dedican al alquiler de robots para el cumplimiento de tareas.
- Disminución de las emisiones de carbono gracias a la eficiencia energética.
- Fabricación de los manipuladores robóticos con materiales sostenibles.

Con el pasar del tiempo los niveles tecnológicos incrementan y con ellos el desarrollo de la robótica, afrontando varios retos como los descritos anteriormente; y solventando las nuevas necesidades que el ser humano tiene (párr. 7,8, 19-22).

Robot Industrial

Tras analizar diversas definiciones para conceptualizar el término "robot", González (2022), concluye que una de las definiciones más ampliamente aceptadas es la proporcionada por la Asociación de Industrias Robóticas (RIA), la cual establece que:

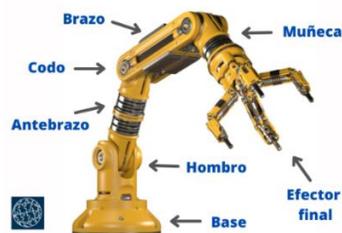
“Un robot industrial es un manipulador multifuncional reprogramable, capaz de mover materias, piezas, herramientas o dispositivos especiales, siguiendo trayectorias variables programadas para llevar a cabo diversas tareas.”(párr. 2).

Un robot industrial está compuesto por articulaciones que enlazan eslabones metálicos, imitando la estructura de un brazo humano. La

Figura 1 muestra las partes constituyentes de un brazo robótico. Para un funcionamiento óptimo de esta máquina, es esencial la presencia de un ordenador que controla el robot al accionar los motores vinculados a cada articulación mediante incrementos precisos. Esta acción se realiza en colaboración con sensores de movimiento que garantizan un desplazamiento preciso y adecuado.

Figura 1

Partes de un brazo robótico industrial.



Nota. Partes constitutivas de un brazo antropomórfico, asemejándolo con las partes del cuerpo humano. Tomado de (Sicma21, 2021).

En el año 2022, InfoPLC a través de su departamento de redacción da a conocer el análisis de los niveles de implementación robótica a nivel mundial exponiendo lo descrito a continuación:

Según el Informe Mundial de Robótica 2022 #WorldRobotics emitido por la IFR (Federación Internacional de Robótica), en el año 2021 se alcanzó un hito sin precedentes con la instalación de 517.385 nuevos robots industriales en fábricas de todo el mundo. Este hecho supone un incremento del 31% en comparación con el año anterior y supera en un 22% el récord de instalaciones registrado en 2018, antes de la pandemia. Además, el informe indica que el número total de robots operativos en todo el mundo ha alcanzado una cifra récord de alrededor de 3,5 millones de unidades.

Para el 2021 Asia se posiciona como el principal mercado para robots industriales, pues acaparó la instalación del 74% de la producción total de robot desplegados en dicho año. China, que es la mayor adoptante de robots en la región, experimentó un crecimiento del 51% en sus instalaciones, representando un total de 268 195 unidades enviadas en 2021. Las instalaciones restantes de robots en todo el mundo fueron desplegadas en este país. Además, el número de acciones operativas de robots en China ha superado la marca de 1 millón de unidades, figurando un aumento del 7%. A continuación, Japón ocupa el segundo lugar con un aumento del 22% de instalaciones (47 182 unidades); Estados Unidos se posiciona en el tercer lugar a nivel mundial con un incremento del 14% en instalaciones robóticas y República de Corea se consolidó como cuarto mercado en ámbitos de instalaciones anuales.

Respecto a los países europeos las instalaciones de robots incrementaron un 24% (84 302 unidades) en el 2021, manteniendo una demanda constante el sector automotriz; mientras que la demanda del sector industrial en general fue de un 51%. Los

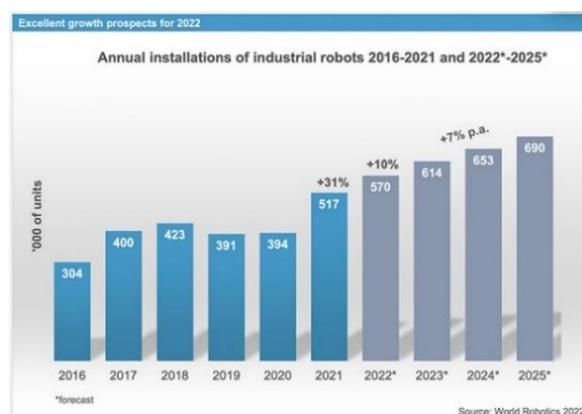
tres países que encabezan las instalaciones de robots son: Alemania con un 28%, Italia con 17% y Francia con 7%.

La elevación de precios en sistemas de abastecimiento de energía, precios intermedios en productos y poca disponibilidad de elementos electrónicos, desafían a todas las ramas de la actual situación económica a nivel mundial. Pero la demanda y ordenes de pedidos de robots industriales han sido mayores; tras la situación económica suscitada después de la pandemia el 2022, fue un año en el que se esperó un crecimiento global del 10% en instalaciones de robots.

A nivel mundial se espera que el auge económico posterior a la crisis pierda algo de fuerza en 2022. Entre 2021 y 2024, se pronostican tasas de crecimiento anual medio de un solo dígito. Es posible que se produzcan pequeñas disminuciones debido a efectos estadísticos, y que la "recuperación" se produzca en 2022 o 2023, pero esto no afectará a la tendencia general de crecimiento. Se prevé que en 2024 se alcance la impresionante cifra de 500 000 unidades instaladas anualmente en todo el mundo, dicho pronóstico se evidencia en la Figura 2.

Figura 2

Gráfico de tasa de crecimiento anual en instalaciones de robots.



Nota. Los datos de esta gráfica son dados por la IFR. Tomado de (InfoPLC, 2022).

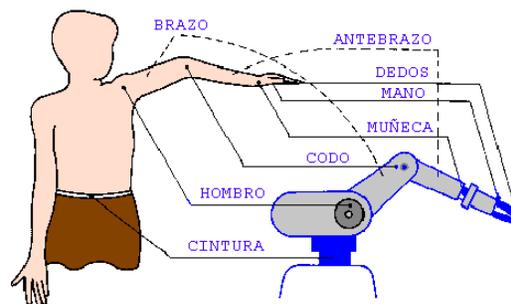
Partes de un robot industrial.

En el año 2020, Cecy argumentó en su publicación:

Un robot industrial consta de: mecánica, engranajes, actuadores, sensores, componentes finales y un controlador. La complejión física en su mayor parte de los autómatas, se asemejan a la constitución anatómica de una extremidad superior del cuerpo humano; la relación estructural de la máquina – ser humano, se ejemplifica en la Figura 3.

Figura 3

Partes constitutivas de un brazo robótico.

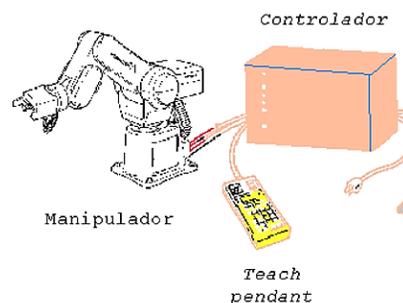


Nota. Similitud entre un robot industrial y las partes constitutivas de una extremidad superior del cuerpo humano. Tomado de (Cecy., 2020).

A continuación, se describe brevemente las partes esenciales de un robot industrial, mismas que se ejemplifican en la Figura 4.

Figura 4

Partes constitutivas de un robot industrial

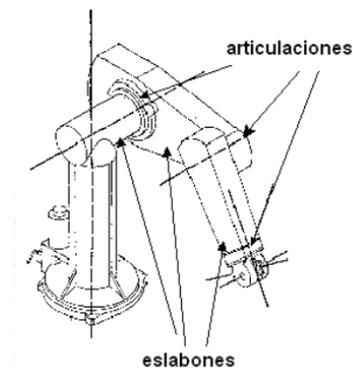


Nota. Componentes de un robot industrial. Tomado de (Cecy., 2020).

- *Manipulador*: componente principal; conformado por eslabones que se encuentran unidos por articulaciones (grados de libertad), que permiten movimientos relativos entre dos eslabones consecutivos (Figura 5).

Figura 5

Partes constitutivas de un manipulador.



Nota. Eslabones y articulaciones de un robot industrial. Tomado de (Cecy., 2020).

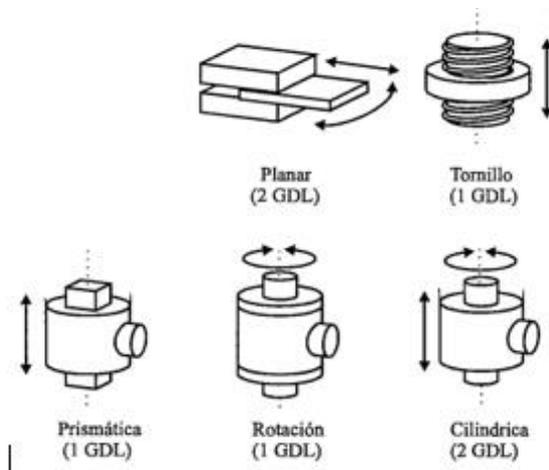
Las articulaciones permiten realizar movimientos lineales (horizontal o vertical) y angulares (debido a articulaciones); este componente de la estructura mecánica da como resultado los grados de libertad que el robot posee, que se enlistan a continuación:

- Cilíndrica (2 GDL)
- Planar (2 GDL)
- Prismática (1 GDL)
- Rotacional (1 GDL)
- Tornillo (1 GDL)

Estas articulaciones se muestran en la Figura 6.

Figura 6

Tipos de articulaciones.

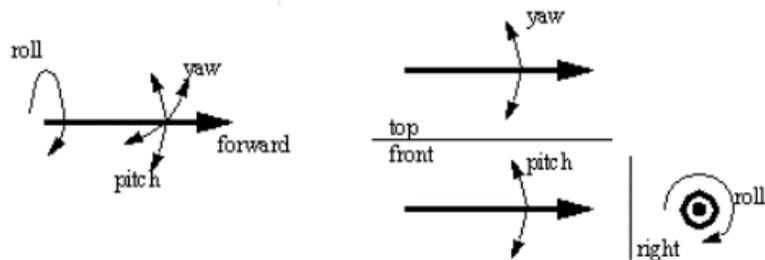


Nota. Tipos de GDL que puede poseer un robot industrial. Tomado de (Barrientos et al., 1997).

La orientación de un eslabón se define por los movimientos roll, pitch y yaw; estos movimientos gráficamente se representan en la Figura 7.

Figura 7

Orientación de los eslabones.

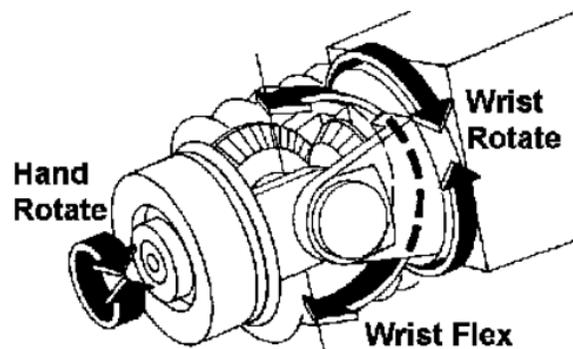


Nota. Representación gráfica de la orientación de los elementos de un eslabón. Tomado de (Cecy., 2020).

La muñeca del manipulador presenta los siguientes movimientos: giro (hand rotate), elevación (wrist flex) y desviación (wrist rotate), estos movimientos se evidencian en la Figura 8.

Figura 8

Movimientos de la muñeca.

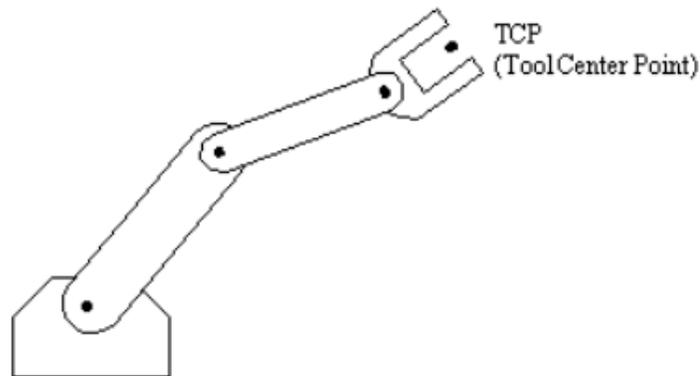


Nota. Movimientos de la muñeca del robot. Tomado de (Cecy., 2020).

Uno más de los componentes del manipulador es denominado como actuador final, que es el que permite la conexión entre el brazo y la muñeca con el objetivo de realizar una tarea en concreto; para las tareas a realizar los elementos terminales se categorizan en: pinzas y herramientas; estos últimos cuentan con un Punto de Centro de Herramienta (TCP, tool center point) que es el punto focal de la herramienta como se visualiza en la Figura 9.

Figura 9

TCP de un brazo robótico.



Nota. Punto focal de la herramienta del brazo robótico. Tomado (Cecy., 2020).

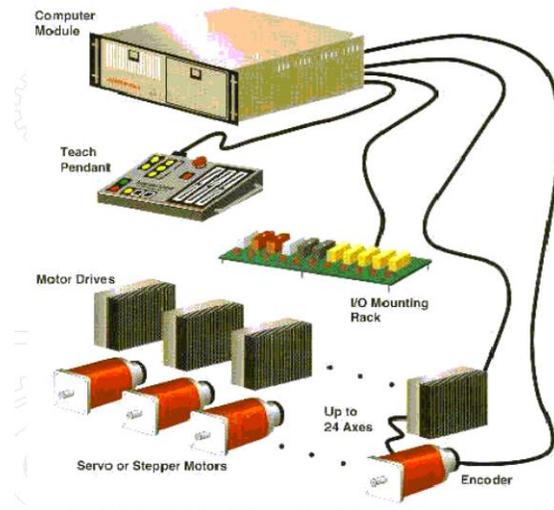
- *Controlador:* responsable de regular los movimientos, acciones, cálculos, procesamiento y almacenamiento de información del operador; está diseñado para enviar y recibir señales a otras máquinas.

Existen varios tipos de controladores, entre ellos se tiene a:

- De posición: opera el control de la posición del componente terminal.
 - Cinemático: control de posición y velocidad.
 - Dinámico: control de las características dinámicas del manipulador y accesorios asociados.
 - Adaptativo: control de la variación de las cualidades del manipulador al modificar la posición.
- *Dispositivos de entrada y salida:* Entre los más utilizados se encuentran el teclado, el teachpendant, la pantalla y los bloques de comandos. Estos dispositivos se presentan en la Figura 10.

Figura 10

Dispositivos de entrada y salida de un robot industrial.



Nota. Dispositivos de entrada y salida de un robot industrial. Tomado de (Cecy., 2020).

Características de los robots industriales

En 2020, Legarreta & Martínez expusieron en su presentación para que un robot sea considerado industrial, éste debe contar con las siguientes características:

- *Grados de libertad:* Se determina por la suma de GDL de las articulaciones que posee un robot, permite giros y desplazamientos para determinar la posición y orientación del elemento terminal del manipulador. Un robot moderno posee 6GDL, 3 para posición y 3 para orientación.
- *Zona de trabajo:* Definida por los GDL y las dimensiones de los componentes del manipulador.
- *Precisión de movimiento:* Se define con la interacción de tres factores.
 - Resolución Espacial: mínimo incremento de movimiento para fraccionar el volumen de trabajo.

- Exactitud: facultad de un robot para colocar la terminación de la muñeca en un punto específico dentro del área de trabajo.
- Repetibilidad: posibilidad para que el robot regrese a un punto programado cuantas veces sean necesarias.
- *Carga a sostener*: aptitud del autómatas para transportar el peso de un objeto con la garra del manipulador.
- *Velocidad de acción*: velocidad máxima que puede conseguir el TCP o articulaciones.
- *Tipos de actuadores*: produce movimiento a las articulaciones y según el tipo de alimentación que tengan, pueden ser oleo-hidráulico, neumático o eléctrico.
- *Nivel de programabilidad*: este controlador brinda la habilidad de supervisar múltiples aspectos operativos, que abarcan:
 - Control de velocidad y aceleración.
 - Saltos de programas condicionales.
 - Gestión de temporizaciones y pausas
 - Edición, modificación, depuración y expansión de códigos
 - Funciones de seguridad integradas.
 - Sincronización con equipos adicionales.
 - Aplicación de lenguajes específicos de robótica.

Tipos de Robot Industriales

Con premisa en la información anteriormente detallada, se realiza un reconocimiento de los distintos tipos de robots industriales existentes.

En el año 2022, Andrade & Curay explican en su trabajo de titulación lo siguiente:

Los autómatas industriales de acuerdo a su estructura y volumen de trabajo que se manifiestan en base a sus movimientos se categorizan en robots clásicos y no clásicos.

Robots Industriales Clásicos.

Dentro de los robots clásicos se tienen robots de tipo cartesiano (PPP), cilíndrico (RPP), esféricos (RRP) y articulados; las categorizaciones antes mencionadas se describen a continuación:

- *Robots Cartesianos (PPP)*: Estos robots se componen de tres ejes cartesianos: X, Y, Z. Cada eje realiza un movimiento prismático o lineal, permitiendo un desplazamiento en direcciones específicas. Este tipo de robot es ampliamente empleado en tareas de manipulación, paletizado, soldadura, impresión 3D, etiquetado y clasificación.
- *Robots Cilíndricos (RPP)*: Los robots cilíndricos exhiben tres movimientos. El primero es rotacional y los otros dos son de naturaleza prismática o lineal. La disposición de estos movimientos otorga al robot un espacio de trabajo que se asemeja a la forma de un cilindro, dando origen a su denominación. Estos robots son utilizados en diversas aplicaciones, beneficiándose de su diseño para ejecutar tareas específicas de acuerdo a su configuración.
- *Robots Esféricos (RRP)*: Constituido por tres movimientos, los dos primeros de índole rotacional y el tercero perteneciente a un movimiento prismático.
- *Robots Articulados*. Estos robots poseen 6 Grados de Libertad (GDL) y presentan una apariencia antropomórfica, es decir, su estructura se asemeja a la morfología humana en cintura, hombro, codo y muñeca. Todos los movimientos que realiza este tipo de robot son de naturaleza rotacional. Debido a su configuración estructural que se asemeja al cuerpo humano, los robots articulados son ampliamente empleados en el ámbito industrial.

Robots Industriales No Clásicos.

Dentro de los robots no clásicos se tienen robots de tipo SCARA, DELTA y COBOTS; las categorizaciones antes mencionadas se describen a continuación:

- *Robots SCARA*. Conocido así por su nombre en inglés, Selective Compliance Assembly Robot Arm; posee de tres hasta cuatro GDL, con dos articulaciones rotativas paralelas a los ejes X, Y; y una o dos articulaciones de carácter prismático perpendicular al eje Z; poseen una gran acogida en el sector industrial debido a la facilidad para su manipulación; comúnmente utilizado para aplicaciones de carga y descarga de alimentos, ensamble de telefonía, entre otros.
- *Robots DELTA*. Constituido por una parte fija encajado al techo y una móvil en la parte inferior del mismo; con movimientos permitidos por los tres grados de libertad que posee. Este tipo de robots son aplicados en procesos sometidos a altas velocidades (clasificación, impresión 3D, ensamble de circuitos).
- *COBOT*: Tras el análisis de varias concepciones en 2022, Robotnik en su publicación presenta el siguiente argumento:

En el año 1996, los profesores J. Edward Colgate y Michael Peshkin desarrollaron los primeros COBOTS como parte de un proyecto auspiciado por General Motors. Estos robots colaborativos fueron patentados bajo la descripción de “un dispositivo y un método para lograr interacción física directa entre un individuo y un manipulador de propósito general controlado por una computadora”(párr. 3).

Aplicaciones y usos de los robots industriales

Una vez clasificados y definidos los robots industriales, en los párrafos consecuentes se da a conocer las aplicaciones a los que éstos se encuentran sometidos. Los robots industriales abarcan varios sectores de producción, para actividades específicas, según un estudio realizado en 2021, UNIR presenta en su publicación las siguientes aplicaciones:

- En el ámbito de la salud, se emplean robots industriales y sistemas automatizados para simplificar la labor de los profesionales en tareas que requieren gran precisión y donde

un error podría tener consecuencias fatales para las personas, como es el caso de las operaciones quirúrgicas.

- En la industria manufacturera, como la del sector automotriz, se busca mejorar la eficiencia y la calidad mediante la automatización de las tareas monótonas y sencillas en la cadena de producción.
- En diversos campos como la agricultura, la seguridad, la defensa, la producción de alimentos, entre otros, cada vez se está haciendo mayor uso de la robótica industrial para llevar a cabo una amplia variedad de actividades y labores.

Una vez analizada la temática de robótica industrial, sus características, tipos de robots y aplicaciones; a continuación, el escrito se centra en el tema robótica colaborativa, características, aplicaciones de los robots colaborativos, esto debido a que el tema del proyecto de titulación en desarrollo trata sobre el control de trayectorias de un COBOT (robot colaborativo), enfocado a la aplicación de empaqueo de productos.

Robótica Colaborativa

Muchos autores expresan su punto de vista sobre la robótica colaborativa, coincidiendo en que “la robótica colaborativa se define como una rama de la robótica que se centra en construir sistemas capaces de compartir espacio, trabajar con personas de forma fiable y segura” (EDS Robotics, 2022, párr. 3), lo que ha contribuido a tener un enfoque más acertado acerca de esta ciencia.

El crecimiento de la robótica como disciplina se encuentra principalmente en el sector industrial, dando pasos agigantados para agilizar los procedimientos, reducir el tiempo y maximizar el uso de recursos, todo en línea con los cambios y necesidades de la sociedad moderna.

En 2022, NCYT Amazings publica sus argumentos acerca de la temática, expresando que:

El campo de la robótica colaborativa está íntimamente relacionado con el modelo Industria 4.0 ya que está diseñado específicamente para ayudar a los operadores en tareas monótonas y repetitivas de diversa complejidad para optimizar los recursos involucrados en el proceso. Esta tecnología está al alcance de cualquier empresa, independientemente de su tamaño.

Los COBOTS, también conocidos como robots colaborativos, son máquinas con una estructura similar a un brazo mecánico. Son livianos y reprogramables, lo que permite que el operador se concentre en tomar decisiones mientras el robot realiza las tareas requeridas. Estas máquinas están equipadas con medios para garantizar la seguridad física del operador. Estos accesorios pueden prevenir colisiones y detenerse inmediatamente si se detecta una anomalía en su área de trabajo, lo que otorga al COBOT un alto nivel de seguridad.

Características de los COBOTS

Existen ciertas similitudes con los robots industriales, pero son más las diferencias y beneficios que presenta este tipo de autómatas; entre ellas se tiene:

- *Compacto*: de tamaño reducido, usado en la mayor cantidad de procesos de producción.
- *Flexible*: a diferencia de los robots industriales que trabajan específicamente con una acción configurada previamente, los cobots tienen la capacidad de rápido aprendizaje para nuevas tareas, permitiendo su uso en distintas etapas de un proceso de fabricación.
- *Móviles*: son portátiles por su tamaño y peso a diferencia de un robot industrial que debe ser instalado en una zona segura y con medidas de protección.
- *Fácil configuración y programación*: pueden ser programados con aplicaciones móviles y software, de manera rápida y sencilla, aptos para trabajar en un pequeño intervalo de tiempo.

- *Económicos*: Inversión económica considerable con respecto a un robot industrial; de menor precio tanto en el ámbito de programación como en la instalación; con el objetivo de recuperar la inversión a corto plazo.

En la información anterior se describen algunas de las características que poseen los COBOTS, dando a conocer su rentabilidad en la realización de distintas tareas. (párr. 1,2,7-11)

Una vez estudiados los conceptos de robótica industrial o tradicional, y la robótica colaborativa; se llega a visualizar varias similitudes enfocadas a mejorar procesos de producción, optimización de tiempos y recursos, adaptabilidad a cualquier campo de trabajo; pero al contrastar estos tipos de robótica se encuentran diferencias muy puntuales, como las mostradas en la Figura 11.

Figura 11

Características de un robot tradicional y un COBOT.

Robot Tradicional		COBOT
Instalación fija repetitiva	1	Instalaciones diferentes
Sin interacción humana	2	Con interacción frecuente
Separación del espacio	3	Espacio compartido
Programación compleja	4	Programación fácil e intuitiva
Rentabilidad en años	5	Rentabilidad < 1 año (mayoría de casos)

Nota. Diferencia de características entre un robot tradicional y un COBOT. Tomado de (CFZ Cobots, 2019).

Ventajas de la robótica colaborativa

En el año 2019, Esneca tras un análisis en la comparación de la robótica tradicional y la robótica colaborativa ha argumentado lo siguiente en su publicación:

La conexión entre los robots colaborativos y la automatización industrial ofrece una serie de beneficios significativos. Según la OCDE (Organización para la Cooperación y el Desarrollo Económico), el mercado español se convertirá en uno de los más automatizados. Esto se debe a la adopción industrial temprana de COBOTS, la facilidad de uso y el costo razonable en comparación con los robots industriales convencionales.

Otra ventaja notable de este tipo de robot es su capacidad para brindar a empresas de todos los tamaños numerosas oportunidades para automatizar procesos peligrosos, repetitivos y monótonos que tradicionalmente han sido realizados por humanos.

Más allá de que un operador ponga en riesgo su puesto de trabajo podrá usar a los robots colaborativos como excusa para mejorar sus destrezas y conocimientos en sus campos de trabajo.

En el año 2022, EDS Robotics presenta en una de sus publicaciones un nuevo argumento:

Las ventajas que se presentan en los entornos industrializados y las PYMES (Pequeñas y Medianas Empresas) automatizadas, se tiene:

- *Alto retorno de la inversión:* los cobots ofrecen un retorno de la inversión mucho mayor que otras herramientas de tecnología industrial, lo que los convierte en un punto de partida único para la automatización.
- *Entorno más saludable:* al realizar tareas que son inconvenientes, repetitivas o inseguras para el operador, los cobots contribuyen a un entorno de trabajo más saludable al reducir el estrés físico y los riesgos asociados.

- *Mejor ambiente de trabajo:* El advenimiento de tecnologías innovadoras motiva a los empleados a aprender cosas nuevas, la contaminación acústica se reduce significativamente durante el funcionamiento del robot, lo que contribuye a un entorno de trabajo más silencioso y eficiente.
- *Rápida ejecución.* Gracias a su facilidad de configuración y uso, los cobots permiten una implementación rápida y sencilla de muchos procesos industriales.
- *Bajo requerimiento de conocimientos técnicos:* los cobots no requieren un alto nivel de conocimientos, lo que democratiza la accesibilidad a los robots colaborativos y facilita su comprensión por una amplia audiencia.
- *Optimización del espacio:* los cobots se adaptan a espacios reducidos y, en muchos casos, no requieren estructuras de protección adicionales, lo que minimiza las limitaciones de espacio en el entorno de trabajo.

Aplicaciones y usos de un robot colaborativo

Los cobots pueden hacer cualquier actividad mecánica que un trabajador realizará con sus manos, con la ventaja de que estas máquinas tienen mayor precisión, no presentan estados de cansancio, ni aburrimiento; a más de contar con la toma de decisiones subjetivas en el trabajo. Es por ello que este tipo de robots pueden ser direccionados a realizar tareas de:

- Pick and Place
- Soldadura
- Alimentación de máquinas
- Seguimiento de trayectoria
- Manipulación de muestras de laboratorio
- Control de calidad
- Empaquetado

- Atornillado de piezas
- Pulido
- Montaje milimétrico
- Paletizado

A las actividades antes mencionadas se les pueden implementar una gran cantidad de tareas específicas de un negocio o sector, siempre que estas últimas sean repetitivas y cuenten con las herramientas adecuadas para cumplirlas.

Fabricantes de robots colaborativos

Del mismo modo que a nivel mundial se cuenta con diversos fabricantes de robots industriales, existen empresas dedicadas al diseño y construcción de robots colaborativos, entre ellos se tiene expuestos algunos modelos en la Tabla 1.

Tabla 1

Empresas fabricantes de robots colaborativos.

Fabricante	Ejemplar	Fabricante	Ejemplar
Universal Robots		FESTO	
FANUC		Doosan Robotics	
ABB		OMRON	

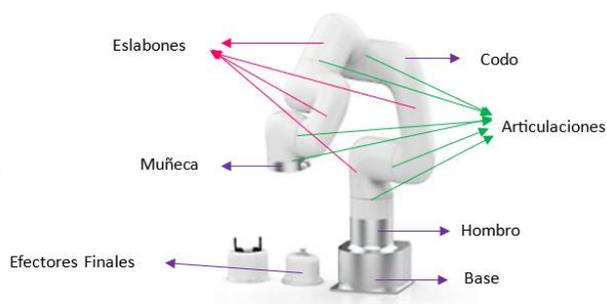
Fabricante	Ejemplar	Fabricante	Ejemplar
YASKAWA		Stäubli	
KUKA		Aubo Robotics	
Techman Robots		DENSO	
UFACTORY			

Partes de un COBOT

La constitución estructural de un robot colaborativo, es semejante al de un robot industrial; pues ambos se componen principalmente de eslabones y articulaciones, permitiendo el movimiento necesario para una aplicación determinada; además ambas configuraciones se asemejan a la constitución antropomórfica de un ser humano, dicha configuración se evidencia en la Figura 12.

Figura 12

Partes constitutivas del COBOT UFACTORY Lite6.



Nota. Partes que constituyen la estructura física del robot colaborativo UFACTORY Lite6.

Tomado de (TEGARA Co., Ltd, 2022).

Área de trabajo de un COBOT

El área o espacio de trabajo de un COBOT depende mucho de la aplicación en la que se utilizará y su fabricante, para ello se muestra en la Figura 13 y la Tabla 2, algunos ejemplares y sus características técnicas, entre ellas se tiene: tamaño, peso, área de trabajo o radio de acción, carga útil, huella (espacio físico que requiere un robot para su instalación), aplicación, velocidad, entre otras.

Figura 13

Características técnicas de robots colaborativos de UFACTORY.

Se muestran dos robots colaborativos de UFACTORY. El primero es el UFACTORY Lite 6, más pequeño y compacto. El segundo es el UFACTORY xArm 6, más grande y con un alcance mayor.

Producto	UFACTORY Lite 6	UFACTORY xArm 6
Peso	9kg	12.2 kg
Alcance	440 mm	700 mm
Carga Útil	800 g ^{max}	5 kg
Repetibilidad	0.2 mm	0.1 mm
Velocidad	500mm/s	1000 mm/s
DOF	6-Axis	6-Axis

Nota. Características técnicas de dos ejemplares de robots colaborativos. Tomado de (KICKSTARTER, 2023).

Donde el robot UFACTORY Lite6 es comúnmente utilizado en automatización de laboratorio, investigación robótica, máquinas de café y pruebas de teclado o pantalla táctil; mientras que el UFACTORY Xarm Lite6 es aplicado en cuidado de máquinas, selección de contenedores, plataforma móvil, automatización de laboratorio e investigación robótica.

Tabla 2

Características técnicas de robots colaborativos de la empresa Universal Robots.

Ejemplar	Tamaño	Huella	Carga Útil	Radio de acción	Peso	Aplicación
 MODELO UR3	Pequeño	128 mm	3 kg	500 mm	11 kg	Fácil integración en centros de producción
 MODELO UR5	Mediano	149 mm	5 kg	850 mm	18.4 kg	Pick & place, ensayos, pruebas de análisis y laboratorio
 MODELO UR10	Grande	190 mm	10 kg	1300 mm		Empaquetado, paletizado y montaje.

Celda Robotizada

Definición

Según la norma UNE-ISO 10218-2:2011, las celdas robotizadas consisten en un conjunto de equipos y componentes que, en combinación con el robot industrial, permiten la realización de tareas de manera ordenada y segura (Amieva, 2021).

Las celdas robotizadas otorgan versatilidad y flexibilidad al agilizar, simplificar y repetir de manera constante diversos procesos de producción sin comprometer la calidad de los productos. Además, garantizan un alto nivel de seguridad para las personas que trabajan en conjunto con los robots. Estas celdas se pueden utilizar en una amplia gama de tareas, como por ejemplo: en la carga y descarga de productos, manipulación de materiales, paletización, soldadura, manufactura, etc. (CENTRIC, 2022).

Etapas de diseño de una celda robotizada

Según Navarro (2020), la creación de una celda robotizada es una actividad mecatrónica que involucra la colaboración de un equipo compuesto por diversos expertos en diferentes disciplinas. Se deben establecer los criterios necesarios para seleccionar los componentes que formarán el sistema completo. Para ello se realizan una serie de pasos "seleccionar el elemento de trabajo final", "diseñar la base del robot", "instalar los elementos de seguridad y protección de la celda de robótica", "desarrollo de control y comunicación sistemas" y "programación de robots".

Modos de funcionamiento

Según *Celdas robotizadas* (s. f.), el concepto de celdas robotizadas surge con el propósito de aumentar la productividad y la calidad de los procesos de fabricación. La mayoría de las celdas robotizadas cuentan con distintos modos de operación, por lo general son tres modos, que se describen a continuación:

- **Modo de programación:** Se utilizan posiciones y trayectorias para marcar los movimientos de los robots. En este modo, la velocidad del robot se reduce considerablemente, aproximadamente al 20% de su velocidad nominal. Además, el dispositivo de control debe incluir mecanismos de parada de emergencia.
- **Modo de ajustes o semiautomático:** Los movimientos se realizan a velocidad real y requiere una atención detallada a las posiciones. En este modo, las medidas de seguridad se ajustan nuevamente a las condiciones de trabajo. Además, la consola de programación debe estar equipada con dispositivos de parada de emergencia.
- **Modo de automático:** Se llevan a cabo los movimientos a velocidad máxima dentro de la celda, después de una verificación de las trayectorias generadas.

Layout

El Layout de un sistema es un procedimiento que requiere varias iteraciones y tiene como objetivo determinar el tipo y la cantidad de robots a utilizar, así como los periféricos necesarios, para el correcto funcionamiento de la celda robotizada, además, establecer su disposición relativa en ésta. Durante este proceso, la experiencia del equipo técnico encargado del diseño es fundamental, también, el uso de herramientas informáticas, como sistemas CAD, simuladores de celdas robotizadas, simplifica en gran medida esta labor (Barrientos et al., 2005).

Es inusual que un robot industrial opere de manera aislada, ya que generalmente forma parte de un proceso de fabricación en el que necesita interactuar con otros elementos, como bandas transportadoras, mesas, herramientas, entre otros. En esta situación, generar el layout desempeña un papel fundamental, puesto que, permite determinar la posición de los periféricos activos y pasivos que actúan en la celda robotizada (Hernández & Vizán, 2015).

Distribución de los equipos en la celda robotizada

Determinar la disposición de los equipos es un procedimiento iterativo que tiene como objetivo establecer la especificación del tipo y la cantidad de robots a utilizar, así como de los elementos activos y pasivos que la conforman, indicando además su disposición relativa, los cuales desempeñan un papel crucial en la celda de trabajo (Trelles, 2021).

- **Periféricos pasivos:** Son componentes o dispositivos que no requieren energía externa para funcionar y que desempeñan un papel en el sistema sin una acción activa. Como, por ejemplo: mesas, cercas de seguridad, soportes, sistemas de fijación, etc.
- **Periféricos activos:** Son fundamentales para la operación y la automatización de la celda robotizada, ya que, son los encargados de realizar tareas y/o controlar otros elementos de la celda de trabajo. Como, por ejemplo: robots industriales, actuadores, sistemas de control, etc.

Disposición del robot en la celda robotizada

Definir la posición del robot dentro de la celda robotizada, según Senao et al. (2004), se puede tener las siguientes disposiciones:

- **Robot en el Centro de la Celda:** En esta disposición, el robot se coloca en el centro de la celda robotizada, rodeado por los demás componentes involucrados en el proceso. Esta estructura es especialmente útil para robots con articulaciones, permitiéndoles aprovechar al máximo su capacidad de movimiento. Es común en aplicaciones de carga y descarga, soldadura por arco, paletización y ensamblaje, donde el robot necesita alcanzar múltiples puntos fijos dentro de su zona de trabajo.
- **Robot en Línea:** Esta disposición es efectiva en sistemas donde las piezas se desplazan, siendo ideal para situaciones en las que la precisión es crucial. Un ejemplo

es la soldadura de carrocerías de vehículos, donde varios robots realizan puntos de soldadura mientras los vehículos avanzan secuencialmente frente a ellos.

- **Robot Móvil:** En algunas situaciones, es beneficioso tener un robot que se mueva a lo largo de una vía predeterminada. Esto es especialmente útil para mantener una posición relativa constante entre el robot y la pieza durante todo el proceso. Una vez completada la tarea, el robot regresa a su posición inicial para el siguiente ciclo de operación.
- **Robot Suspendido:** Esta disposición es común en robots tipo pórtico, que se colocan sobre el área de trabajo. Ofrece ventajas como una utilización más eficiente del espacio, ya que el robot puede acceder a puntos en su eje vertical.

Características del sistema de control de la celda robotizada

Una vez que los elementos operativos de la celda y su disposición en la planta han sido establecidos, el siguiente factor crucial en la definición de la celda robotizada es la especificación del sistema de control, como señalan (Odorico et al., 2007).

Según Montoya (2019), es de vital importancia establecer un sistema de control eficiente que pueda desempeñar diversas funciones, tales como:

- Control individual de cada una de las máquinas, transportadores y dispositivos que integran la celda, incluyendo los robots.
- Sincronización del funcionamiento de los distintos dispositivos entre sí.
- Detección, manejo y posible recuperación de situaciones anómalas que puedan surgir durante la operación.
- Optimización de la operación conjunta de los dispositivos de la celda, con la posibilidad de distribuir dinámicamente funciones para evitar inactividades y acciones innecesarias.

- Interfaz con el usuario, proporcionando información pertinente para que el estado del sistema sea comprensible en todo momento y permitiendo al operador acceder, con restricciones apropiadas, al funcionamiento del sistema.
- Interfaz con otras celdas para lograr sincronización y optimización en un sistema de fabricación flexible compuesto por varias celdas.
- Interfaz con un sistema de control superior que supervisa y actualiza programas cuando se produce un cambio en la producción.

Estas funciones pueden ser necesarias en distintos niveles para controlar una celda robotizada y serán implementadas en hardware que debe ser definido y dimensionado adecuadamente. En casos donde la celda es relativamente sencilla, el controlador del robot podría asumir las funciones de control tanto de sus propios ejes como de otros dispositivos, utilizando sus entradas y salidas digitales o analógicas. No obstante, si la celda incluye múltiples dispositivos, algunos de los cuales tienen sus propios controladores, será necesario establecer una estructura jerárquica donde un elemento central se comuniquen con los demás controladores, según la propuesta de (Montoya, 2019).

Es importante considerar el tamaño adecuado del sistema de control de la celda de trabajo, ya que esto afectará los costos de implementación, desarrollo y operación. No se debe subestimar la importancia del dimensionamiento correcto del control de celda, ya que su impacto será directo en el rendimiento general del sistema como describe (Barrientos et al., 2005).

Modos operativos de los cobots en una celda robotizada

Es fundamental establecer un entorno en el cual el cobot pueda llevar a cabo sus tareas y coexistir de manera segura con los seres humanos. Por lo que, es necesario comprender, crear y verificar este entorno para garantizar la seguridad tanto de los trabajadores humanos

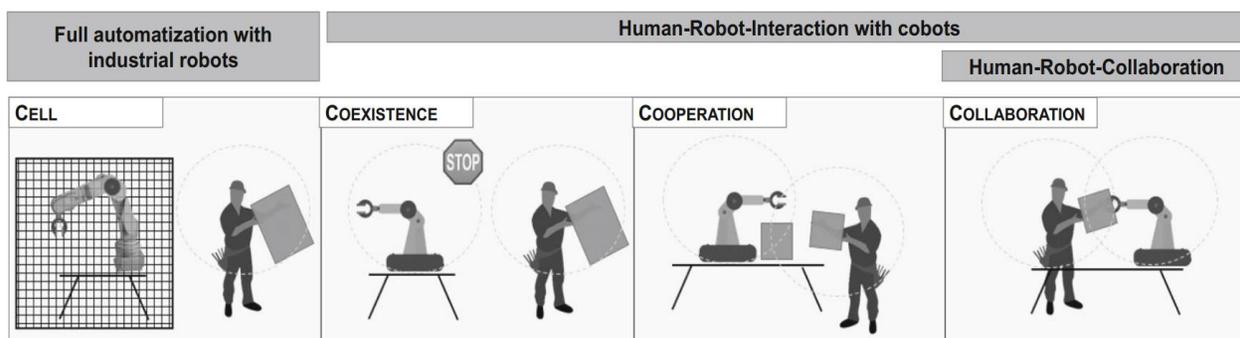
como del propio cobot, para lo cual, se han creado normas para su correcta implementación. (Gisginis, 2021). Por ejemplo, las normas ISO 10218-1/2, identifican modos de colaboración de los cobot de manera que puedan coexistir con los humanos de forma segura (Villani et al., 2018).

En la automatización completa de procesos con robots industriales, en las celdas de trabajo el ser humano y el robot no comparten un espacio de trabajo común. Pero la colaboración humano-robot, se refiere a un tipo especial de interacción que se caracteriza por el contacto físico directo entre un cobot y un ser humano, según la definición ISO 8373:2012. En consecuencia, mientras que robots industriales convencionales requieren una separación física de los humanos por motivos de seguridad, los cobots permiten varias opciones de interacción que van de la coexistencia a la colaboración (Kopp et al., 2021).

Existen tres maneras en las que los cobots y los humanos pueden interactuar dentro de una celda robotizada, como se puede observar en la Figura 14, también, se detalla cada una de estas formas de interacción entre humanos y cobots.

Figura 14

Modos de interacción entre cobots y los humanos.



Nota. Modos de interacción entre cobots y humanos. Tomado de (Kopp et al., 2021).

- **Coexistencia.** El cobot y el ser humano ocupan el mismo entorno, pero ninguno de ellos invade el área de trabajo del otro, prescindiendo de la necesidad de una barrera física que separe al robot del ser humano (Salvador, 2019).
- **Cooperación.** Los humanos y cobots comparten un espacio de trabajo común, la cooperación se caracteriza por pasos de trabajo secuenciales y no requiere necesariamente contacto físico para realizar el trabajo. El humano y el cobot están presentes en el mismo espacio de trabajo, pero no al mismo tiempo (Kopp et al., 2021).
- **Colaboración.** Tanto el cobot como el humano colaboran de forma activa en la misma zona de trabajo, pero en este escenario en particular, el cobot cuenta con la capacidad de responder en tiempo real a los movimientos del operario (Salvador, 2019).

Seguridades para el trabajo en una celda robotizada

Elementos de Seguridad de una celda robotizada

En el año 2023, Valdez realiza una publicación en la que argumenta que una celda robotizada consta de los siguientes componentes que la vuelven segura:

- *Barreras de acceso a la celda:* Se colocan delimitantes alrededor de la celda, impidiendo el acceso a cualquier persona.
- *Dispositivo de intercambio de piezas:* impide que el operador no se ponga en peligro al momento de poner o recoger piezas situadas en el área de trabajo.
- *Movimientos condicionados:* cuando el operario requiere el ingresar al área de trabajo, previamente el robot será configurado para no realizar movimiento alguno en ese momento.
- *Zona de reparación:* Se tendrá en cuenta la presencia de áreas de mantenimiento y reparación fuera del área de trabajo.

- *Señalización apropiada*: la celda debe contener la señalética apropiada del trabajo o línea robotizada.
- *Paro de emergencia*: debe contener un interruptor de seguridad, mismo que bloquea la acción de trabajo del robot en caso de una emergencia (p. 11).

Normativa Legal para el Trabajo en las Celdas Robotizadas con COBOTS

En el año 2007, Lascano realiza una investigación donde se da a conocer las normativas legales que intervienen en el diseño, construcción y mantenimiento de las celdas y sistemas robóticos, donde argumenta:

- La norma internacional ISO 10218:1992 establece las reglas y directrices para la regulación de sistemas robóticos. Esta norma aborda diferentes aspectos relacionados con la seguridad y el diseño de dichos sistemas. Incluye secciones como el análisis de seguridad, que permite identificar riesgos y posibles fuentes de peligro o accidentes. Esta norma se encarga del diseño y producción, con análisis detallados acerca de los sistemas robotizados, considerando aspectos mecánicos, ergonómicos y de control.
- La norma estadounidense ANSI/RIA R15.06-1992 a pesar de ser corta, contiene elementos importantes. Por ejemplo, en la sección de definición de riesgo, se incluyen varios subtemas relacionados con la probabilidad de ocurrencia de accidentes y la gravedad de los mismos. Estos subtemas abordan la seriedad y severidad de los accidentes, considerando el nivel de daño físico que puede sufrir una persona, según la experiencia del operador y la frecuencia con la que se encuentra en la zona de peligro.
- Las regulaciones EN 775 en Europa y UNE-EN 775 en España se basan en las normativas internacionales ISO 10218:1992 y EN 775, respectivamente. Estas regulaciones se centran en el diseño de robots industriales y celdas de trabajo robotizadas, así como en su uso, reparación y seguridad durante las etapas de mantenimiento (p. 5).

Seguridad en robots colaborativos

En el año 2020, Vergara realiza una publicación presentando su argumento sobre la seguridad en robots colaborativos, donde indica:

Los COBOTS, también conocidos como robots colaborativos, se comercializan como dispositivos incompletos. Para su uso seguro, recae en la responsabilidad de la empresa integrarlos con otros equipos, maquinarias o herramientas, llevar a cabo una evaluación de riesgos y emitir una declaración de conformidad CE, considerando el conjunto como una única entidad. Aunque no existe una normativa específica para la seguridad de los COBOTS, se aplica la norma UNE-EN ISO 10218-2 dentro de la directiva de máquinas para las configuraciones robóticas.

La evaluación de riesgos se basa en la norma ISO/TS 15066, que establece los criterios de seguridad y los parámetros operativos de los COBOTS. Es de suma importancia ajustar los COBOTS según los parámetros específicos de cada aplicación, tomando en cuenta aspectos como el peso de las piezas, las superficies de contacto y la velocidad del proceso, a fin de asegurar la seguridad en interacciones con personal humano.

Los COBOTS deben cumplir con los requisitos estipulados en la directiva de máquinas y llevar el marcado CE. Cada aplicación precisa ser evaluada por profesionales versados en los requisitos normativos pertinentes. Resulta fundamental que los integradores, programadores y operarios de COBOTS tengan la capacidad de configurar aplicaciones inequívocas para los obreros, garantizando así la seguridad en las operaciones empresariales (párr. 2, 3).

Seguridad en el espacio de trabajo colaborativo

En el año 2020, Lange argumenta en su publicación que a nivel internacional existen estrategias de seguridad para trabajar con cobots, para aplicaciones colaborativas de manera segura y eficaz, por ello:

En entornos colaborativos, es de suma importancia llevar a cabo la identificación y el mapeo exhaustivo de todos los equipos adicionales presentes en el espacio de trabajo, en conjunto con los robots. Estos equipos deben someterse a una evaluación de los riesgos potenciales, y es esencial la instalación de sensores de seguridad para prevenir posibles lesiones y daños.

Si existen equipos no colaborativos que son considerados seguros, pero podrían requerir dispositivos de seguridad adicionales, como aquellos utilizados para manipulación de materiales, mecanizado, pinzas, actuadores y otras máquinas, es factible integrar de manera sencilla dispositivos de seguridad en aplicaciones que involucren robots colaborativos.

Una variedad de soluciones se encuentra disponibles para salvaguardar los espacios de trabajo colaborativos. En áreas con riesgos bajos y de acceso libre, se pueden emplear escáneres de área y alfombrillas de seguridad. En espacios cerrados o con riesgos más elevados y operaciones a alta velocidad, las barreras ópticas de seguridad y los interruptores de seguridad son alternativas viables. Para situaciones que impliquen peligros activos o actividades con riesgos, los operadores pueden contar con interruptores de "hombre muerto" que automáticamente desactivan el sistema si el usuario deja de aplicar presión sobre ellos.

A fin de asegurar una seguridad óptima en las operaciones colaborativas, los fabricantes deben validar la seguridad de sus aplicaciones con robots en todas las fases de la operación. Aunque existen directrices para evaluar la seguridad de un robot mientras trabaja junto a un operador, todavía pueden existir riesgos potenciales, como activaciones accidentales o descargas eléctricas, incluso cuando el robot se encuentra en estado inactivo (p. 1).

Tipos de operación colaborativa

Para 2016, tecnopl.com realiza una redacción de robótica y tecnología, titulada "Seguridad de COBOTS: Cuatro Tipos de Operación Colaborativa" donde argumenta la existencia de estos tipos de operación mismos que se detallan a continuación:

- *Parada monitorizada de seguridad:* El sistema de parada monitorizada de seguridad en contextos de colaboración en el entorno laboral implica que el robot se detiene antes de que el operador humano se exponga a cualquier tipo de peligro. Solo en ausencia de un operador, el robot puede moverse como si no fuera colaborativo.
- *Guiado Manual:* En cuanto a la operación de guiado manual, el operador humano utiliza un dispositivo manual para transmitir órdenes de movimiento al sistema del robot. En esta modalidad, el operador tiene un control directo sobre el funcionamiento del robot. Aunque el operador está involucrado en el proceso, esta forma de operación sigue siendo considerada automática y no manual.
- *Monitorización de velocidad y separación:* Para utilizar el procedimiento de monitorización de velocidad y separación, se implementan dispositivos de seguridad externos para los COBOTS, como un escáner de seguridad.
- *Limitación de potencia y fuerza:* Este enfoque se basa en la probabilidad de que los seres humanos interactúen con un sistema de robot en movimiento. Por lo tanto, al evaluar los riesgos, se considera el impacto que esto pueda tener en el cuerpo humano. Con el propósito de prevenir posibles molestias o lesiones, la aplicación limita tanto la carga que el robot puede transportar como su velocidad, esta limitación puede resultar en una velocidad demasiado baja para ser eficaz en aplicaciones de alto riesgo (párr. 2,3-6,7).

Software de programación de robots industriales.

Lenguajes de programación de robots

Según Comíns, 2018 a inicios de la década de los 50, comenzó la utilización de la computación para el control de máquinas-herramienta, lo que dio origen al Control Numérico Computarizado (CNC). El desarrollo inicial del CNC tuvo lugar en el Instituto Tecnológico de

Massachusetts, donde se automatizó por primera vez una fresadora de gran tamaño. La distinción entre el lenguaje CNC y el lenguaje de programación robótica no siempre se encuentra claramente definida. La diferenciación parece depender más del dispositivo en el que se ejecuta el programa: si se trata de una máquina-herramienta, se considera CNC, y si involucra un brazo con múltiples ejes, se considera un lenguaje robótico.

A finales de los años 50 y principios de los años 60, se comenzaron a desarrollar lenguajes de programación específicos para robots. Uno de los primeros y más primitivos lenguajes fue WAVE, que fue creado en 1970 en la Universidad de Stanford, se considera ampliamente como el primer lenguaje de programación de robots, sentó las bases para los lenguajes de programación robótica que se utilizan en la actualidad.

A lo largo de cuatro décadas, hubo un crecimiento significativo en el desarrollo de lenguajes de programación robótica. Durante este período, tanto universidades e institutos de investigación como empresas especializadas en el campo de la robótica contribuyeron al diseño y desarrollo de estos lenguajes.

Es importante resaltar la notable longevidad de los lenguajes de programación, en particular aquellos desarrollados por las empresas fabricantes de robots, se presenta en la Tabla 3 un resumen de los lenguajes más significativos utilizados en la programación de robots industriales.

Tabla 3

Lenguajes de programación de robots industriales.

LENGUAJE	SIGNIFICADO	UNIVERSIDAD/ FABRICANTE	CARÁCTERÍSTICA
WAVE (1973)	-	Stanford	Manejo de piezas mecánicas. Acoplamiento de bomba de agua. Ensamblaje

LENGUAJE	SIGNIFICADO	UNIVERSIDAD/ FABRICANTE	CARACTERÍSTICA
AL (1974)	Assembly Language	Stanford	Manipulación de piezas mecánicas. Sistema de incremento de programas Pointy. Robots Unimation.
RAPT (1978)	Robot Automatically Programmed Tool	Sistemas de Toulouse (Universidad de Edimburgo)	Montaje
VAL I (1979) VAL II (1983)	Versatile Assembly Language	Unimation	Robot Puma de Unimation
MCL	Manufacturing Control Language	Cincinnati Milacron	Apto para la programación de robots "off-line".
LRP	Live Robot Programming	ACMA	Programar un robot o simulador que se encuentra ejecutando el código mientras éste está siendo escrito.
AML (1979)	A Manufacturing Language		
FUNKY	-	IBM	Pertenece al nivel de descripción de las gammas de montaje actuales.
MAPLE	-		
AUTOPASS (1977)	Automatic Parts Assembly System		
V+ (1989)	-	ADEPT	Usado en los robots Adept y Staübli
KAREL	-	FANUC	Basado en PASCAL y soporte multitarea.

LENGUAJE	SIGNIFICADO	UNIVERSIDAD/ FABRICANTE	CARACTERÍSTICA
KRL	Kuka Robot Language	KUKA	Basado en el estándar IRL, descrito en la norma DIN 66312.
RAPID (1994)	Robotics Application Programming Interactive Dialogue	ABB	Uso de estructuras preestablecidas para la configuración del robot y las cualidades de la herramienta.

Nota. Lenguajes de programación de robots industriales a lo largo de la historia. Tomado de (Yebra, 2022).

A partir del año 2000, en lugar de seguir publicando nuevos lenguajes de programación específicos, comenzaron a surgir numerosos "frameworks" o marcos de trabajo. Estos frameworks, plataformas o paquetes de software se enfocan en el desarrollo de programas para robots. Por lo general, estos frameworks admiten varios lenguajes de programación, ofrecen abstracción para diferentes tipos de robots, proporcionan entornos de desarrollo y simulación, así como bibliotecas y componentes adicionales. Esta evolución hacia los frameworks ha brindado a los programadores de robótica una mayor flexibilidad y opciones para desarrollar aplicaciones robóticas (Comíns, 2018).

Métodos de programación

Es fundamental disponer de instrucciones que los robots puedan interpretar y ejecutar. Estas instrucciones actúan como un lenguaje de comunicación entre el programador y el robot. Estos métodos de programación están diseñados para simplificar el proceso de programación por parte del usuario, permitir una interpretación y ejecución rápida de las tareas programadas en los robots industriales. El procedimiento de programación se refiere al conjunto de pasos

requerido para instruir el sistema robótico o la secuencia utilizada para determinar la lista de acciones que debe ejecutar el robot (Guaraca & Ochoa, 2015).

Dentro de la programación de robots, existen dos métodos principales: la programación en línea (PEL) y la programación fuera de línea (PFL). Cada uno de estos métodos tiene sus propias características distintivas, que se describen a continuación.

Programación en Línea

Esta metodología de programación es el procedimiento original y más utilizado. El operador guía el robot manualmente a través de todos los puntos de la secuencia a medida que se registran. Esta secuencia de pasos se repite luego hasta que no se produzcan errores. La serie se puede guardar con los puntos clave más importantes o incluso como una serie completa de puntos (Razo, 1999). Algunas formas de programación son:

- **Programación por aprendizaje directo.** El aprendizaje directo es un método de programación en el que se manipula la muñeca o el brazo del robot para llevarlo a los puntos deseados durante la fase automática. Es una forma sencilla de programación, sin complicaciones, que permite realizar trayectorias rápidas para procesos simples en la industria. El controlador del robot registra automáticamente las señales de los sensores de posición coincidentes en todos los puntos donde se mueve el robot. Sin embargo, este método tiene limitaciones de edición, ya que se deben guardar o definir muchos puntos para crear una ruta continua, lo que puede generar espacios vacíos si se reduce (Yebra, 2022).
- **Programación mediante un dispositivo de enseñanza.** En este método, es necesario establecer parámetros como velocidad, tipo de trayectoria y precisión deseada para alcanzar los puntos específicos. A diferencia del aprendizaje directo, los usuarios no necesitan tener conocimientos de programación, ya que pueden familiarizarse con el uso de los elementos de enseñanza del dispositivo. Con esto, pueden realizar

secuencias, modificaciones y otras acciones dentro del programa. Sin embargo, este enfoque está más orientado a aplicaciones simples (Guaraca & Ochoa, 2015).

Programación fuera de línea

La programación fuera de línea presenta varias ventajas, como la capacidad de reutilizar código, flexibilidad para realizar modificaciones y reducción del tiempo de inactividad del sistema durante la etapa de programación. Además, al reprogramar las tareas de un robot que forma parte de un proceso de fabricación, solo se requiere una breve interrupción mientras los nuevos programas se cargan en las computadoras de control de las estaciones de trabajo. Esto puede resultar en un importante ahorro de tiempo y una reducción en los costos operativos (Sánchez, 2020). Se divide en:

- **Programación Textual explícita.** Se basa en la elaboración de un texto compuesto por instrucciones o sentencias que permiten indicar las tareas que el robot debe realizar. En este caso, no es necesario la intervención directa del robot, ya que se utiliza un lenguaje de programación específico para definir las acciones del brazo manipulador. Mediante la programación textual, el operador no define las acciones del robot de manera manual, sino, que se calculan en el programa utilizando las instrucciones textuales apropiadas (Yebra, 2022).
- **Programación gráfica.** La programación gráfica utiliza elementos visuales que se pueden ajustar mediante el uso de parámetros en lugar de texto. Estos elementos gráficos, como bloques o diagramas de flujo, se combinan para crear programas. Una ventaja significativa de este enfoque es que simplifica en gran medida el proceso de programación. Sin embargo, la desventaja es una menor flexibilidad en comparación con la programación textual. Un ejemplo conocido de interfaz gráfica de programación es LEGO Mindstorms, donde se construye un robot con legos y se programa utilizando

una interfaz gráfica en un ordenador o tableta. Al combinar los bloques y especificar sus parámetros, el usuario puede definir el comportamiento del robot (Hader, 2021).

- **Fuera de línea – gráfica:** Este enfoque implica la combinación de programación textual con una metodología manual, realizada en un entorno simulado de un robot a través de una interfaz gráfica interactiva (Razo, 1999).

En la programación fuera de línea, el usuario tiene acceso a modelos virtuales del robot y su entorno en una computadora. Estos modelos son geoméricamente idénticos al robot y a su entorno físico real. Una vez que se han establecido y programado los movimientos del robot, es posible visualizar una animación del robot ejecutando la tarea en la computadora. Esto permite verificar el éxito de los movimientos planificados antes de su ejecución en el robot real. En resumen, la programación fuera de línea proporciona una herramienta virtual para validar y optimizar los movimientos del robot antes de llevarlos a cabo en la realidad (Sánchez, 2020).

Existen dos tipos de software utilizados en la programación fuera de línea que son:

- **Software específico de cada fabricante.** Lo que significa que el programa solo es compatible con los robots fabricados por ese fabricante en particular. Estos programas ofrecen un entorno de simulación completo y también incluyen módulos para ejecutar tareas especiales, como soldadura, seguimiento de bandas transportadoras y sincronización de ejes de movimiento, entre otros (Gonzalez, 2019).
- **Software de propósito general:** Proporciona un entorno de simulación que es compatible con robots de varios fabricantes. Estos programas ofrecen modelos cinemáticos y dinámicos de diferentes robots y, en algunos casos, también incluyen post procesadores que generan instrucciones de programación específicas (González, 2019).

En resumen, los softwares de programación fuera de línea específicos del fabricante se limitan a robots de ese fabricante en particular, mientras que los programas de propósito general son compatibles con robots de múltiples fabricantes. En la Tabla 4 se presenta un listado de los desarrolladores más reconocidos de programas enfocados en la programación fuera de línea.

Tabla 4

Principales desarrolladores de programas para la programación fuera de línea.

Software de acceso genérico		Software proporcionado por fabricantes	
Software	Compañía	Software	Compañía
Delmia	Dassault Systems	Robotstudio	ABB
RobCAD	Technomatrix	Motosim	Yaskawa-Motoman
Robomaster	Robomaster	Kuka-Sim	KUKA
Robsim	Camelot	Roboguide	Fanuc
Cosimir	Festo	Melfa Works	Mitsubishi
RoboDK	RoboDK	PC ROSET	Kawasaki
Drag&Bot Studio	Drag&Bot	UFACTORY Studio	Ufactory

Nota. Listado de programas de propósito general y desarrollado por fabricantes para programación fuera de línea. Tomado de (Gonzalez, 2019).

Métodos actuales de programación de cobots

Al igual que los robots industriales, los cobots pueden ser programados de forma en línea y fuera de línea, pero poseen diferentes métodos de programación en comparación con los mencionados anteriormente, como, por ejemplo:

- **Programación por demostración.** Es uno de los enfoques más simples para programar un cobot, en este método, el operador tiene la capacidad de mover físicamente el cobot hasta la posición deseada mientras mantiene presionado un botón específico y sujeta el efector final de manera firme. Una vez que se logran las configuraciones deseadas, es posible guardar los puntos de control y crear rutinas

básicas. Este enfoque se destaca por su facilidad de uso, pero está limitado a la ejecución de tareas muy simples. La mayoría de los fabricantes de cobots del mercado emplean este método debido a su simplicidad e intuición en comparación con otros métodos de programación (Commare, 2021).

- **Sistemas de aprendizaje.** En este tipo de sistemas, el robot aprende a realizar una tarea a partir de ejemplos utilizando métodos de aprendizaje automático, como redes neuronales o aprendizaje por refuerzo. Esto permite que el robot pueda programarse a sí mismo, adaptando su comportamiento y mejorando su rendimiento. Este enfoque es especialmente útil para refinar secuencias de tareas ya programadas y maximizar la eficiencia del robot.
- **Sistemas instructivos.** Es un método que permite controlar el robot en tiempo real utilizando lenguaje natural o gestos. Sin embargo, es necesario enseñar estos comandos de alguna manera previamente, como a través de la programación por demostración. Aunque este enfoque ofrece una mayor intuición, puede comprometer la fiabilidad del sistema. Por lo tanto, en aplicaciones industriales, las interfaces gráficas de usuario y las soluciones hápticas son más adecuadas, ya que proporcionan una mayor precisión y confiabilidad (Hader, 2021).

Software para programación de cobots.

La programación genérica de robots se refiere al soporte de un sistema de programación fuera de línea, para escribir un programa de robot en un lenguaje o modelo de programación para el que existen medios para convertir dicho programa en instrucciones o comandos de programa específicos del proveedor. La programación genérica de robots permite la interconexión del software con diferentes tipos de cobots (Ionescu, 2020).

Se mencionó anteriormente que cada fabricante de cobots tiene su propio software de programación, pero el acceso a estas herramientas suele ser costoso. Como resultado, cada

vez es más común que empresas especializadas en ingeniería de software desarrollen aplicaciones que integren componentes de diferentes fabricantes en un solo entorno. Estas aplicaciones utilizan lenguajes de programación como JAVA, C++, Python, entre otros, para simular el comportamiento de los cobots, incluso generando programas que pueden cargarse directamente en el controlador real del cobot. Esto permite a las empresas acceder a soluciones de programación más asequibles y flexibles (Rodríguez, 2021).

A continuación, se detallan los más populares:

- **SIMUMATIK.** Esta es una plataforma de emulación fundamentada en la nube que proporciona a los usuarios la capacidad de desarrollar sistemas y procesos con una alta precisión, como se puede observar en la Figura 15; cuenta con un editor de componentes que permite a los usuarios crear cualquier tipo de elemento que se necesite. Utilizando un modelo de datos abierto, los usuarios pueden modelar todos los aspectos importantes de un componente, como su geometría y su física. Para establecer el comportamiento físico y dinámico del componente, se utiliza código Python. Simumatik sigue un enfoque basado en componentes naturales, lo que significa que se pueden crear modelos digitales de sensores, actuadores, controladores e incluso robots, y conectarlos entre sí para construir un sistema, tal como se haría en el mundo real (Rodríguez, 2021).

Figura 15

Interfaz de la plataforma de emulación SIMUMATIK.



Nota. Interfaz del software SIMUMATIK. Tomado de (SIMUMATIK, 2020).

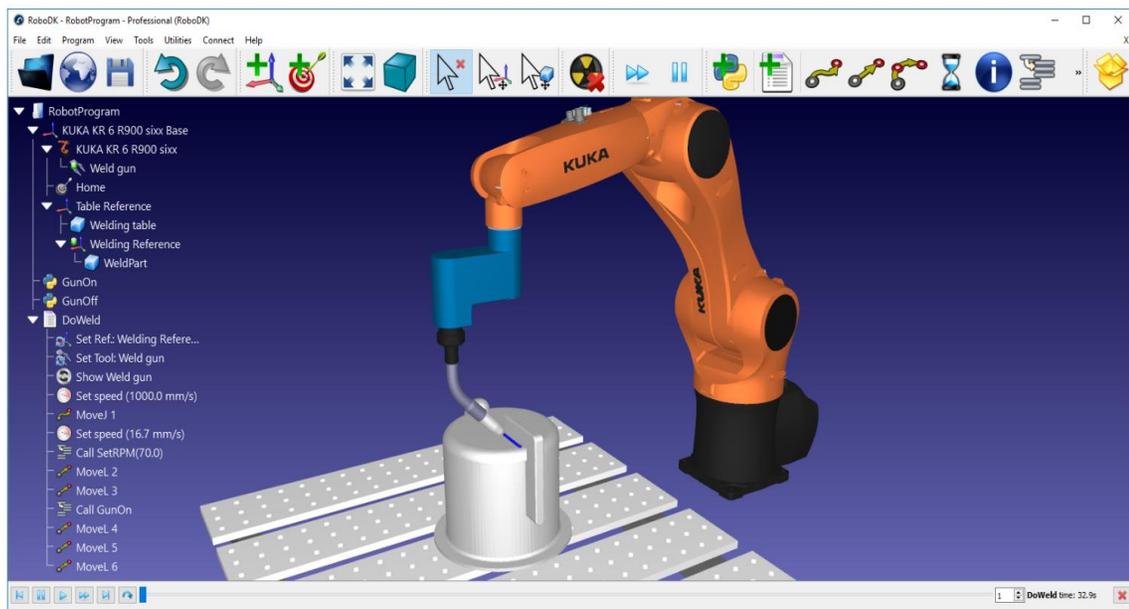
- **RoboDK.** Es un software que ofrece la posibilidad de programar de forma offline una amplia gama de modelos de robots industriales, incluyendo robots colaborativos, todo en un mismo entorno. Este software resulta especialmente útil en entornos de fabricación industrial, ya que permite la creación, simulación y generación de programas de robots en un modo offline, como se puede ver en la Figura 16. Sin embargo, una limitación de esta herramienta es que proporcionan un conjunto restringido de estructuras de control, ya que, RoboDK carece de sentencias condicionales básicas como "if - then - else" (Ionescu, 2020).

RoboDK ha sido diseñado para facilitar la creación de celdas robotizadas completas y funcionales sin la necesidad de escribir código. Esto se logra mediante una interfaz que proporciona una programación gráfica, donde los usuarios pueden utilizar diferentes iconos disponibles en la barra de herramientas para realizar diversas acciones, como programar objetivos, movimientos lineales, articulares y circulares, establecer tiempos de espera y cambios en señales digitales, entre otras funciones. Sin embargo, para aquellos usuarios que estén más familiarizados con la programación convencional o deseen realizar acciones más específicas, RoboDK también ofrece una interfaz de programación que permite escribir programas en el lenguaje Python (Cuello, 2020).

Uno de los aspectos más destacados de RoboDK es su capacidad para realizar una simulación precisa en 3D del robot y el entorno de trabajo; lo cual permite realizar un análisis y estudio de diversos escenarios en una celda robotizada antes de su implementación en la línea de producción, evitando errores futuros. El software también cuenta con la capacidad de generar automáticamente el código del robot en lenguajes de programación ampliamente utilizados, como KRL (KUKA Robot Language), RAPID y Python, lo que simplifica el proceso de programación y se ajusta a las preferencias y necesidades de los usuarios (Ginés, 2023).

Figura 16

Interfaz de la plataforma de emulación RoboDK.



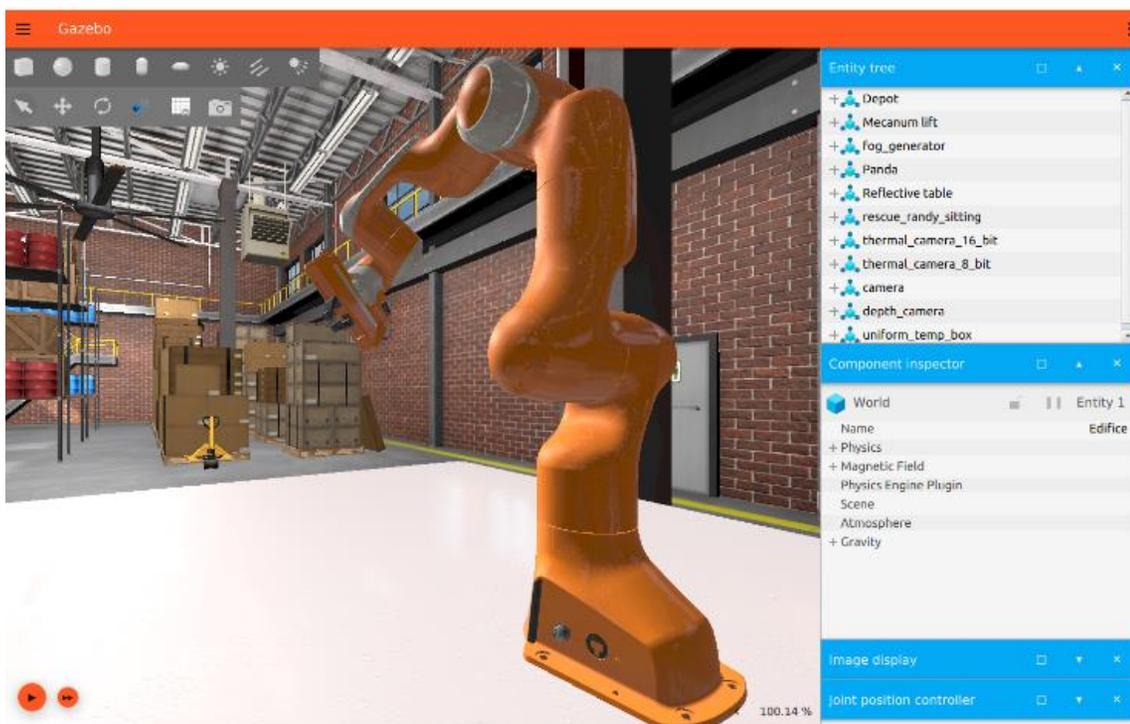
Nota. Interfaz del software RoboDK. Tomado de (RoboDK, 2023).

- **GAZEBO.** Se trata de una plataforma que brinda una variedad de capacidades que incluyen la evaluación de algoritmos, la creación de diseños de robots, la realización de pruebas de regresión y la capacitación de sistemas de inteligencia artificial (IA) mediante simulaciones que representan ambientes realistas, como se muestra en la Figura 17. Gazebo dispone de un motor de simulación física sólido, presentación gráfica

de alta calidad e interfaces tanto gráficas como de programación que resultan convenientes. Lo destacado de esta plataforma es que es gratuita. Además, brinda la posibilidad de desarrollar plugins personalizados para controlar los robots, los sensores y el entorno. También se dispone de una amplia biblioteca de robots, y en caso de que el robot deseado no esté disponible, siempre se puede construir uno propio utilizando el formato de descripción de robot (Rodríguez, 2021).

Figura 17

Interfaz de la plataforma de emulación GAZEBO.



Nota. Interfaz del software GAZEBO. Tomado de (Gazebo, s. f.).

Capítulo III

Selección de componentes

Introducción

En este capítulo se describe el proceso para la selección de componentes de la celda robotizada didáctica para el control de trayectorias en un brazo robótico antropomórfico mediante programación offline con software educativo, para la aplicación al empaquetado de productos, en el laboratorio de robótica industrial de la Universidad de las Fuerzas Armadas ESPE sede Latacunga, se realiza un análisis de las necesidades y requisitos del usuario. Este proceso tiene como objetivo identificar con precisión los sistemas y subsistemas que constituirán el proyecto en cuestión.

Planificación del diseño

Definición de las necesidades

En la Tabla 5, se detallan las necesidades y especificaciones funcionales que se requieren para el diseño de la celda didáctica.

Tabla 5

Necesidades en función de los usuarios.

No.	Necesidades
1	Resistente.
2	Fácil manejo.
3	Fácil mantenimiento.
4	Interacción con el usuario.
5	Empaquetar de productos.
6	El material de la mesa debe ser metálico.
7	Alimentación continua de productos.
8	Programación fuera de línea.
9	Área de trabajo en función del cobot.
10	Nuevo modelo de brazo antropomórfico.

Definición de las especificaciones técnicas

En la Tabla 6, se detallan las necesidades expresadas de manera técnica, para la celda robotizada enfocada al empaquetado de productos.

Tabla 6

Especificaciones técnicas de la celda robotizada.

No.	Especificaciones Técnicas
1	Robusto.
2	Operación Simple.
3	Arquitectura Modular.
4	Modo de operación cooperativo.
5	Control de trayectorias.
6	Componentes normalizados.
7	Velocidad constante de la banda.
8	Software educativo para programación fuera de línea.
9	Celda elaborada con normas.
10	Selección de un modelo de COBOT.

La escala de correlaciones a utilizarse para la elaboración de la matriz QFD se detallan en la Tabla 7.

Tabla 7

Escala de correlaciones.

Descripción	Valor
Relación fuerte	5
Relación media	3
Relación baja	1
Ninguna relación	0

Matriz QFD

Basándose en las necesidades y especificaciones técnicas de la celda robotizada para empaquetado de productos se desarrolla la matriz QFD mostrada en la Tabla 8.

Tabla 8

Matriz QFD del proyecto (Quality Function Deployment)

Necesidades	Especificaciones										
	Importancia	Robusto	Operación Simple	Arquitectura Modular	Modo de operación cooperativo	Control de trayectorias	Componentes normalizados	Velocidad constante de la banda	Software educativo para programación fuera de línea.	Celda elaborada con normas	Selección de un modelo de COBOT
Resistente	4	5	1	1	3	0	1	0	0	5	1
Fácil manejo	1	1	5	3	1	3	1	3	5	1	3
Fácil mantenimiento.	2	1	1	5	0	0	3	1	0	3	3
Interacción con el usuario.	3	0	3	0	5	5	0	5	5	1	5
Empaquetar de productos.	4	0	0	1	3	5	1	3	5	5	5
El material de la celda debe ser metálico.	1	3	0	0	0	0	0	0	0	5	0
Alimentación continua de productos.	3	0	0	1	3	5	0	5	5	3	3
Programación fuera de línea.	5	0	5	1	1	5	0	3	5	3	5
Área de trabajo de $1 m^2$ aproximadamente	1	0	1	0	0	0	0	3	0	5	5
Nuevo modelo de brazo antropomórfico	5	1	3	3	5	1	0	1	5	5	5
Dificultad		2	2	3	3	3	3	3	5	5	5
Resultados del como		31	61	44	79	83	15	70	105	109	112

Especificaciones	Necesidades										
	Importancia	Robusto	Operación Simple	Arquitectura Modular	Modo de operación cooperativo	Control de trayectorias	Componentes normalizados	Velocidad constante de la banda	Software educativo para programación fuera de línea.	Celda elaborada con normas	Selección de un modelo de COBOT
Ponderación absoluta	62	122	132	237	249	45	210	525	545	560	
Ponderación relativa (%)	2,31	4,54	4,91	8,82	9,27	1,67	7,82	19,54	20,28	20,84	
Orden de importancia	9	8	7	5	4	10	6	3	2	1	

Tomando en cuenta los resultados de la Tabla 8, a continuación, se detallan las especificaciones técnicas según su orden de importancia y el porcentaje obtenido de cada una de ellas, como se puede evidenciar en la Tabla 9, de forma que se puedan establecer los sistemas y subsistemas que conforman el proyecto.

Tabla 9

Resultados de la matriz QFD.

Especificación	Porcentaje	Orden de Importancia
Selección de un modelo de COBOT.	20,84%	1
Celda elaborada con normas.	20,28%	2
Software educativo para programación fuera de línea.	19,54%	3
Control de trayectorias.	9,27%	4
Modo de operación cooperativo.	8,82%	5
Velocidad constante de la banda.	7,82%	6
Arquitectura Modular.	4,91%	7

Especificación	Porcentaje	Orden de Importancia
Operación Simple.	4,54%	8
Robusto.	2,31%	9
Componentes normalizados.	1, 67 %	10

Obtenidos los resultados de la Tabla 9, se definen los sistemas y subsistemas que se deben desarrollar en el proyecto, así como las funciones que deben ser implementadas (Tabla 10)

Tabla 10

Sistemas y subsistemas del proyecto.

Sistema	Subsistema	Función
Celda robotizada	Diseño y construcción de la banda transportadora	Transporte continuo de objetos
	Diseño y construcción de la estructura de la celda	Soporte del cobot y de sus componentes
	Selección de los componentes de la celda	Brindar seguridad y origen para la programación fuera de línea
Cobot	Selección del cobot	Aplicación de empaquetado de productos
	Selección del software educativo	Programación fuera de línea del cobot
Control	Tarjeta de control	Control de los sistemas, adquisición de datos, y envío de señales.
	Verificar Trayectorias	Verificación de la programación realizada

Selección de robot antropomórfico.

Al automatizar un proceso en específico, el equipo técnico encargado de esta tarea tiene la responsabilidad de seleccionar el robot más apropiado en función de la aplicación a desarrollar, para hacerlo, se debe elegir uno entre la amplia oferta de robots disponibles en el

mercado, en función de las características requeridas, se debe buscar siempre un equilibrio adecuado entre el precio y las prestaciones ofrecidas (Barrientos et al., 2005).

Características de selección.

Teniendo en cuenta la aplicación de empaquetado a desarrollarse, se definen las siguientes características con las que debe contar el COBOT (Tabla 11).

Tabla 11

Características deseables para la selección del robot.

Característica	Valor	Unidad
Alcance	400 a 500	<i>mm</i>
Grados de libertad	6	#
Repetibilidad	0.1 a 0,2	mm
Velocidad nominal	400 a 600	$\frac{mm}{s}$
Carga útil	0.7 a 1.5	<i>Kg</i>
Modo de programación	On Line – Off Line	-
Peso	12	<i>Kg</i>
Alimentación	110	<i>VAC</i>
Costo	\$3500 a \$5000	\$

Opción 1: Universal Robots - UR3e

La empresa Universal Robotics cuenta con el modelo de COBOT UR3e, mismo que se ajusta a las características requeridas para la aplicación de empaquetado a desarrollarse. En la Tabla 12 se enumeran las características de este robot colaborativo, en la Figura 18 se presenta el modelo del brazo robótico.

Tabla 12

Características del cobot Universal Robots - UR3e.

Característica	Valor	Unidad
Alcance	500	<i>mm</i>
Grados de libertad	6	#

Característica	Valor	Unidad
Repetitividad	0.03	mm
Velocidad nominal	1000	$\frac{mm}{s}$
Carga útil	3	<i>Kg</i>
Modo de programación	On Line Off Line	-
Peso	11	<i>Kg</i>
Alimentación	100 - 240	VAC
Costo	23 000	\$

Figura 18

Cobot de la empresa Universal Robots modelo UR3e.



Nota. Cobot UR3e. Tomado de (Universal Robots, 2023).

Opción 2: FANUC CR-4iA

La empresa FANUC dispone del cobot CR-4iA, el cual se acopla con los requerimientos de la celda robotizada a implementarse. A continuación, en la Tabla 13 se enumeran las características del cobot, en la Figura 19 se ejemplifica este modelo.

Tabla 13

Características del cobot FANUC CR-4iA.

Característica	Valor	Unidad
Alcance	550	<i>mm</i>
Grados de libertad	6	#
Repetitividad	0.01 mm	mm
Velocidad nominal	2 000	$\frac{mm}{s}$
Carga útil	4	<i>Kg</i>
Modo de programación	On Line Off Line	-
Peso	48	<i>Kg</i>
Alimentación	200 – 230	VAC
Costo	30 000	\$

Figura 19

Cobot de la empresa FANUC modelo CR-4iA.



Nota. Cobot CR-4iA. Tomado de (Fanuc, 2023).

Opción 3: UFACTORY Lite 6

La compañía UFACTORY ofrece en su catálogo de robots colaborativos el modelo UFACTORY Lite6, mismo que cumple con los requerimientos para la aplicación de empaquetado. A continuación, en la Tabla 14 se enlistan las características del cobot, en la Figura 20 se representa al robot colaborativo.

Tabla 14*Características del cobot UFACTORY Lite 6.*

Característica	Valor	Unidad
Alcance	440	<i>mm</i>
Grados de libertad	6	#
Repetitividad	0.2	mm
Velocidad nominal	500	$\frac{mm}{s}$
Carga útil	1	<i>Kg</i>
Modo de programación	On Line Off Line	-
Peso	9	<i>Kg</i>
Alimentación	110	VAC
Costo	3 000	\$

Figura 20*Cobot de la empresa UFACTORY modelo Lite 6.**Nota. Cobot Lite 6 Tomado de (Ufactory, 2021).*

Selección del cobot

Considerando las especificaciones técnicas de cada una de las opciones descritas anteriormente, en la Tabla 15, se asigna un peso a los criterios de selección y su respectiva calificación.

Tabla 15

Selección del brazo robótico colaborativo para la celda robotizada.

Aspecto Importante	Peso Asignado	Universal Robots		FANUC		UFACTORY	
		UR3e		CR-4iA		Lite 6	
		Calif.	Ponde.	Calif.	Ponde.	Calif.	Ponde.
Alcance	0,15	10	1,5	9	1,35	9	1,35
Grados de libertad	0,06	10	0,6	10	0,6	10	0,6
Repetividad	0,12	8	0,96	10	1,2	3	0,36
Velocidad nominal	0,05	8	0,4	10	0,5	5	0,25
Carga útil	0,1	9	0,9	10	1	8	0,8
Modo de programación	0,08	10	0,8	10	0,8	10	0,8
Peso	0,03	6	0,18	3	0,09	10	0,3
Alimentación	0,01	7	0,07	7	0,07	10	0,1
Costo	0,4	4	1,6	3	1,2	10	4
Total	1		7,01		6,81		8,56
Selección			No		No		Si

De acuerdo a los resultados de la Tabla 15, la propuesta ganadora es el robot colaborativo de la empresa UFACTORY, modelo Lite 6, el cual ofrece las prestaciones suficientes para el desarrollo de la celda robotizada con programación fuera de la línea con el cual se implementará una aplicación de empaquetado de productos.

Efactor final

El cobot Ufactory lite 6 previamente seleccionado, posee dos efectores finales, una piza y una ventosa eléctrica, debido a la aplicación de empaquetado que se realizará en la celda robotizada, se selecciona la ventosa como efector final del robot.

Vacuum Gripper Lite 6

La pinza de vacío tiene la capacidad de aspirar de manera dinámica objetos planos y lisos con una carga útil de hasta 0,5 kg. Cuenta con tres ventosas que pueden seleccionarse según el tamaño de la superficie del objeto. Se recomienda que la superficie de succión sea lisa, si esta condición no se cumple pueden existir fugas de aire, lo que resulta en una menor firmeza en la aspiración (Ufactory, 2021b). En la Tabla 16, se enumeran las características de la ventosa, en la Figura 21 se representa dicha herramienta.

Tabla 16

Características del Vacuum Gripper Lite 6.

Característica	Variable	Unidad
Voltaje nominal	24	VDC
Peso	171,6	g
Fuerza de succión	60	KPa
Carga útil	0,5	Kg
Funcionamiento	eléctrico	-

Figura 21

Vacuum Gripper Lite 6 de la empresa UFACTORY.



Nota. Vacuum Gripper de la empresa UFACTORY. Tomado de (Reichelt, s. f.).

Selección de software de programación offline.

El software a utilizarse tiene como requerimiento crear el entorno de ejecución donde se pueda controlar, estudiar y analizar trayectorias, a más de poderse visualizar el COBOT UFACTORY Lite 6 y los componentes necesarios de la celda para la aplicación de empaquetado; también se considera la posibilidad de una programación offline.

Opción1. Simulador Robótico GAZEBO

Las características del simulador para robótica GAZEBO se describen en la Tabla 17.

Tabla 17

Características del software de simulación GAZEBO.

Característica	Descripción
Gratuito	Software de código abierto
Compatibilidad con sistema operativo	Linux; MAC; Windows.
Simulación de Dinámica	Acceso a motores de física y alto rendimiento.
Gráficos 3D avanzados	Uso de OGRE (Object Oriented Graphics Rendering Engine) para una presentación realista del entorno.

Característica	Descripción
Sensores y ruidos	Generación de datos de sensores, opcionalmente con ruido.
Herramientas de líneas de comando	Fácil introspección y control de la simulación
Complementos	Desarrollo de complementos personalizados; con acceso directo a la API de GAZEBO.
Modelos de robots	PR2, Pioneer DX, iRobot Create y TurtleBot, creación de un robot propio con SDFFormat.
Transporte TCP/IP	Simulación mediante servidores remotos basado en socket utilizando Google Protobufs.

Nota. Descripción de las características del software GAZEBO. Tomado de (Gazebo, s. f.).

Opción 2. Simulador SIMUMATIK

En la Tabla 18 se detallan las características del software SIMUMATIK.

Tabla 18

Características del software de simulación SIMUMATIK.

Característica	Descripción
Simulación	Sistemas industriales completos. Simulación en tiempo real.
Biblioteca	Componentes y accesorios industriales predefinidos.
Herramientas de diseño	Diseño de sistemas y diagramas de circuitos con una diversidad de herramientas.
Interfaz gráfica intuitiva	Facilidad de localización de herramientas y componentes para la creación y edición de sistemas.
Análisis de rendimiento	Herramientas para análisis de rendimiento en sistemas.
Compatibilidad	Compatibilidad con PLC para la ejecución de tareas.
Integración con entorno CAD	Importación y exportación de diseños desde y hacia ambientes CAD.
Soporte técnico y actualización	Disponibilidad periódica.

Nota. Descripción de las características del software SIMUMATIK. Tomado de (Simumatik, s. f.).

Opción 3. Simulador RoboDK

Las características del software RoboDK se exponen en la Tabla 19.

Tabla 19

Características del software de simulación RoboDK.

Característica	Descripción
Simulación de robots industriales y cobots	Visualización y ejecución programas de manera virtual.
Programación fuera de línea	Creación y generación de programas para robots industriales y cobots.
Compatibilidad de robots	Diferentes marcas y modelos de robots industriales y cobots.
Biblioteca de componentes	Biblioteca de accesorios y componentes robóticos.
Generación de trayectorias de herramientas	Generación automática de trayectorias de herramientas optimizadas para tareas específicas.
Integración con entorno CAD	Disponibilidad de software de diseño CAD.
Importación y exportación de programas	Importar y exportación de códigos de programación de los varios ejemplares en diferentes formatos.
Interfaz gráfica intuitiva	Facilidad de programación y simulación de entornos robóticos.
Soporte y comunidad	Acceso a soporte técnico, conocimientos y experiencias de otros usuarios.

Nota. Descripción de las características del software RoboDK. Tomado de (RoboDK, 2023).

El resultado de la elección del software se muestra en la

Tabla **20**, tras el análisis comparativo de los softwares antes mencionados.

Tabla 20

Parámetros de selección de software de programación offline.

Característica sustancial	Peso Asignado	GAZEBO		RoboDK		SIMUMATIK	
		Calif.	Pond.	Calif.	Pond.	Calif.	Pond.
Programación offline	0.2	0	0	10	2	0	0
Librería para COBOT UFACTORY Lite6	0.2	0	0	10	2	0	0
Lenguaje de programación Python	0.1	10	1	10	1	10	1
Compatibilidad con sistema operativo Windows	0.1	5	0.5	10	1	10	1
Interfaz gráfica amigable	0.1	8	0.8	8	0.8	8	0.8
Integración de sistemas CAD	0.1	7	0.1	8	0.8	8	0.8
Fácil manipulación y aprendizaje	0.1	7	0.7	8	0.8	7	0.7
Control de trayectorias	0.1	5	0.5	9	0.9	2	0.2
Total	1		3.6		9.3		4.5
Selección			No		Si		No

En la

Tabla 20, en concordancia con los requerimientos del software especificados para la aplicación de empaquetado la propuesta ganadora es RoboDK, el cual permite el desarrollo de programación fuera de línea, el control de trayectorias, cuenta con la librería para comunicación

directa del software del COBOT UFACTORY Lite 6, es compatible con el sistema operativo Windows, permite la interacción en tiempo real con el COBOT; garantizando un pleno desarrollo del objetivo planteado del proyecto de titulación.

Capítulo IV

Diseño y Construcción de la Celda

El capítulo describe el proceso de diseño y construcción de la celda robotizada con un robot colaborativo UFACTORY Lite 6 y accesorios para la aplicación de empaquetado de productos.

Parámetros de diseño

El área de trabajo del COBOT seleccionado es de 440 mm de radio en el plano XY, el área de trabajo destinado para la aplicación de empaquetado es la mitad de la envolvente horizontal del robot, en donde se coloca una banda transportadora y un sistema de sujeción regulable para la zona de depósito de los productos.

Las dimensiones máximas del sistema de sujeción establecidas en función de área de trabajo del robot son de 200 mm de largo por 140 mm de ancho y, como dimensiones mínimas de 130 mm de largo por 90 mm de ancho, la finalidad de la matriz adaptable es mantener siempre una posición estable para el proceso de almacenamiento; facilitando el empaquetado de productos, ya que el robot traslada los artículos desde un punto inicial a uno final.

El componente principal de la celda robotizada es el cobot UFACTORY Lite 6, utilizado para trasladar los productos desde un punto de alimentación a una zona de almacenamiento; para realizar esta actividad se considera: la respuesta del robot ante la señal de entrada de un sensor de modo que pueda depositar un producto en una posición específica y la capacidad de carga útil del efector final.

Se requiere el diseño y la implementación de una banda transportadora encargada de proveer los productos al robot.

Analizando los parámetros de diseño de la celda robotizada para la aplicación de empaquetado de productos se requiere de 3 dispositivos indispensables para cumplir con dicha tarea:

- Cobot UFACTORY Lite 6
- Banda transportadora
- Sistema de sujeción regulable

Diseño de la celda robotizada

Determinados los componentes que conforman la celda robotizada, y el área de trabajo del robot, se definen los elementos activos y pasivos que complementan el funcionamiento de la celda para la aplicación de empaquetado.

Elementos activos

En la Tabla 21, se enlistan los elementos activos de la celda.

Tabla 21

Elementos activos que conforman la celda robotizada.

Elemento	Descripción
COBOT UFACTORY LITE 6	Alcance en el eje Z de -165 mm a 683.5mm, la velocidad máxima del efector final es de 500mm/s, el peso de todo el brazo robótico es de 9kg.
VACCUM GRIPPER LITE 6	El peso de la ventosa es de 171.6 g, una fuerza de succión de 60Kpa.

Elementos pasivos

Se definen a continuación, en la Tabla 22, los elementos pasivos que conforman la celda robotizada.

Tabla 22

Elementos pasivos que conforman la celda robotizada.

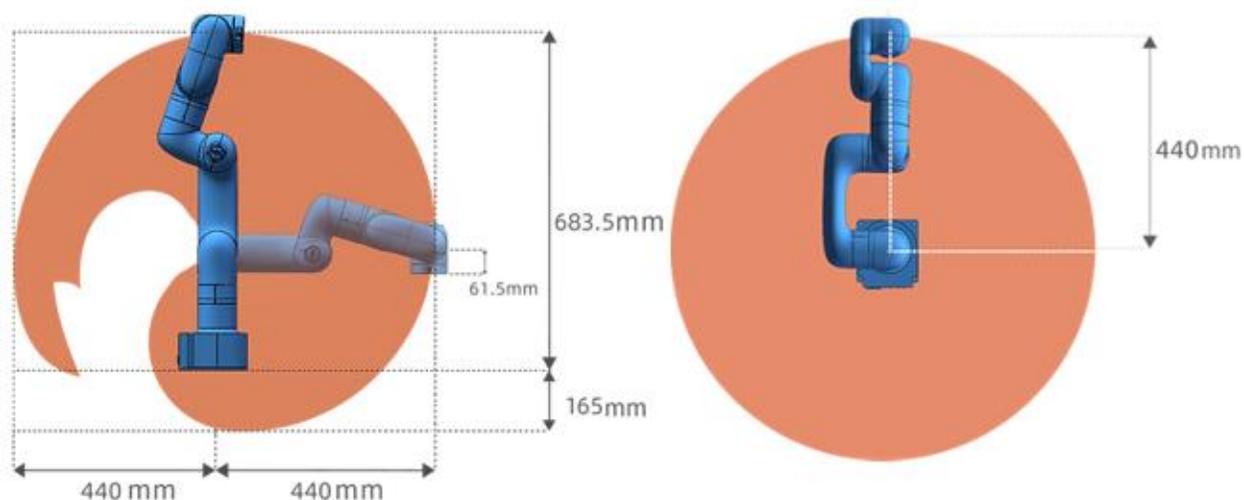
Elemento	Descripción
Mesa	Cuenta con el área requerida para disposición de los componentes de la celda
Banda Transportadora	Transporta los productos desde un punto a otro con la finalidad de automatizar un proceso.
Sistema de sujeción regulable.	Delimita la zona de almacenamiento de la aplicación de empaquetado.
Tablero de control	Almacenamiento de las fuentes de alimentación.

Disposición de la Celda Robotizada

La Figura 22, muestra las dimensiones y el área de trabajo del cobot, lo que permite definir la disposición de los elementos de la celda robotizada, de forma que se pueda evitar colisiones o daños a personas y/o equipos.

Figura 22

Área de trabajo del COBOT Lite 6.

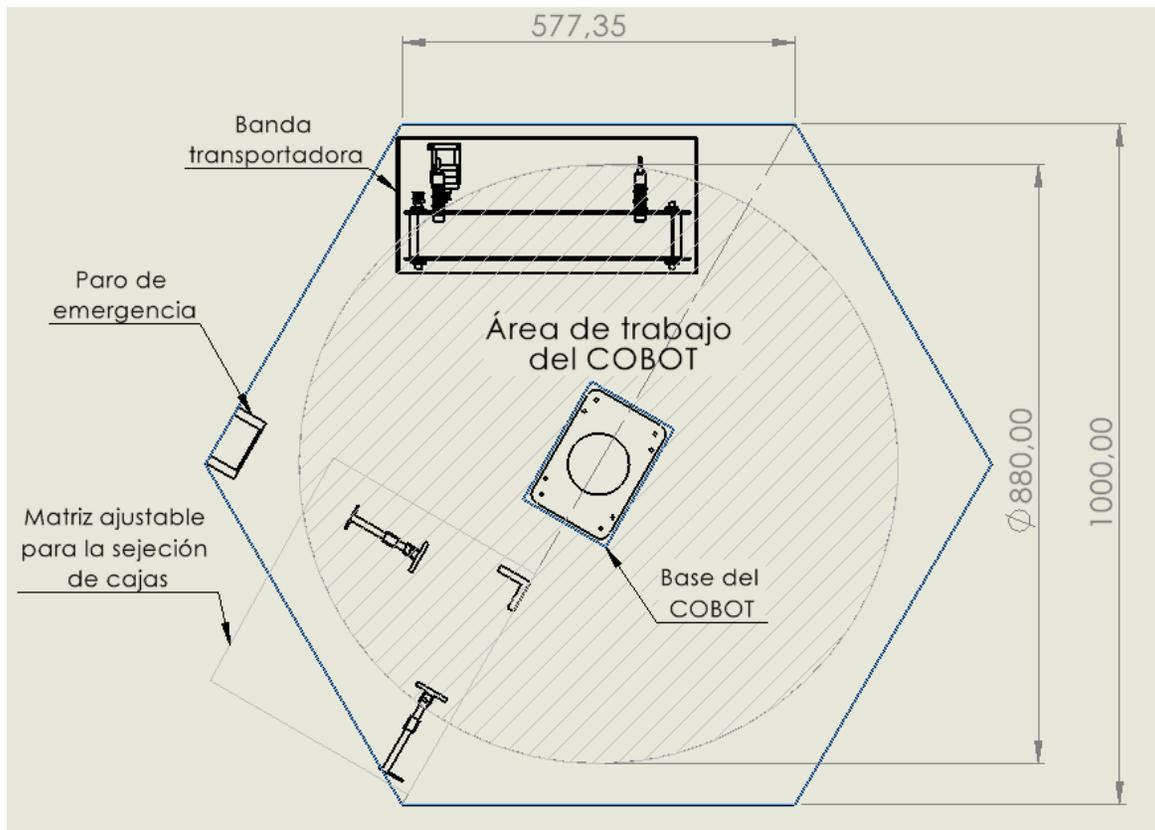


Nota. Área de trabajo del COBOT Lite 6, en el plano XY y en el eje Z. Tomado de (Ufactory, 2021b).

En función de la aplicación de empaquetado la disposición del robot en la celda robotizada es centralizada, en la que los componentes se distribuyen alrededor del robot; el modo de operación de la celda robotizada es cooperativa entre el cobot y el usuario, el operador coloca los productos en un extremo de la banda transportadora de manera secuencial, la banda los traslada hacia el otro extremo, un sensor envía la señal al controlador del robot para la ejecución de la trayectoria programada para el empaquetado de productos. En la Figura 23, se muestra la distribución centralizada de la celda.

Figura 23

Distribución centralizada de la celda.

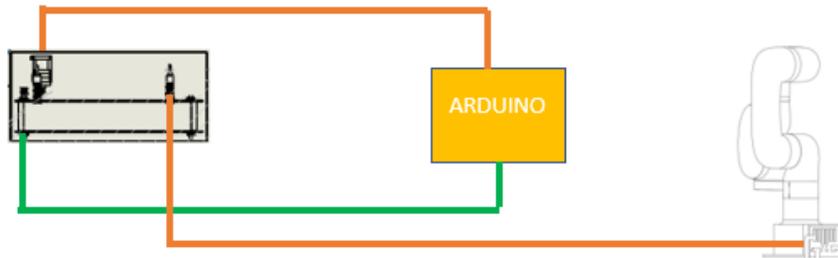


Sistema de control

El sistema de control utilizado en la celda robotizada para la aplicación de empaquetado de productos es un sistema descentralizado, puesto que el robot y la banda transportadora actúan de forma independiente uno del otro, como se puede observar en la Figura 24.

Figura 24

Sistema descentralizado de control



Seguridades de la celda

Los elementos de seguridad disponibles en la celda de robotizada son:

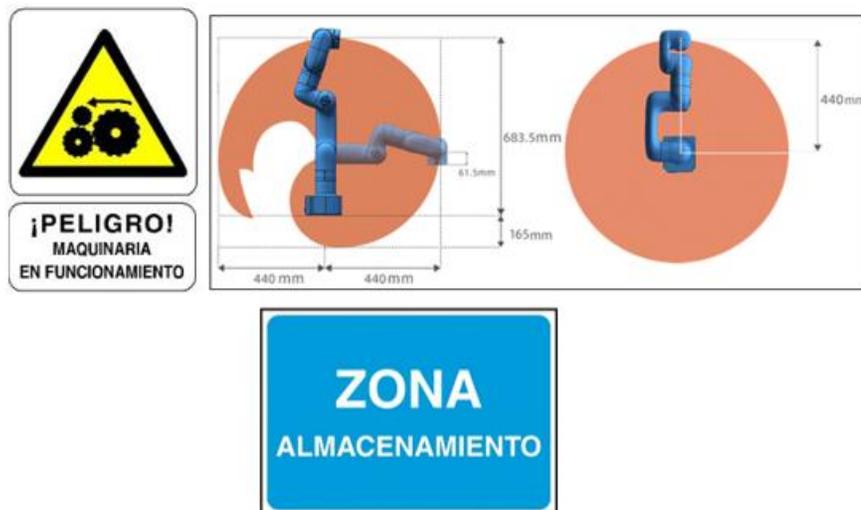
- **Paro de emergencia:** En caso de un incidente o un comportamiento inusual en el brazo robótico, resulta vital accionar el interruptor designado para emergencias (Figura 25). Esta acción interrumpe el movimiento del brazo y lo detiene brevemente. Dicho interruptor está colocado en una zona fuera del área operativa del robot. Al presionar dicho interruptor, la alimentación del brazo robótico se desconectará en un lapso de tan solo 300 milisegundos (Ufactory, 2021b).

Figura 25

Botón paro de emergencia.



- **Señalética de seguridad:** Cumple con la función de alertar a los usuarios de un posible riesgo, las señaléticas implementadas en el proyecto se muestran en la Figura 26.

Figura 26*Señalética de seguridad.***Diseño de la Mesa**

La norma NTE INEN 1 641 determina el uso de la mesa en diferentes posiciones, ya sea sentado, de pie o en una combinación de ambas; la posición en la cual se utiliza la mesa es la de sentado con una altura fija, que bajo norma debe estar en el rango de 740 ± 20 mm. El diseño de la mesa se visualiza en las Figura 27, Figura 28 y Figura 29.

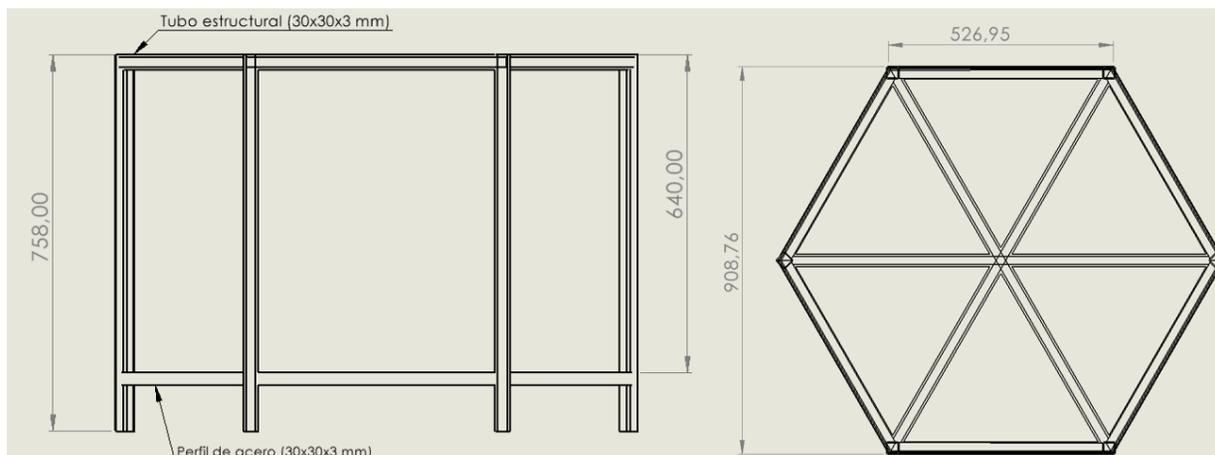
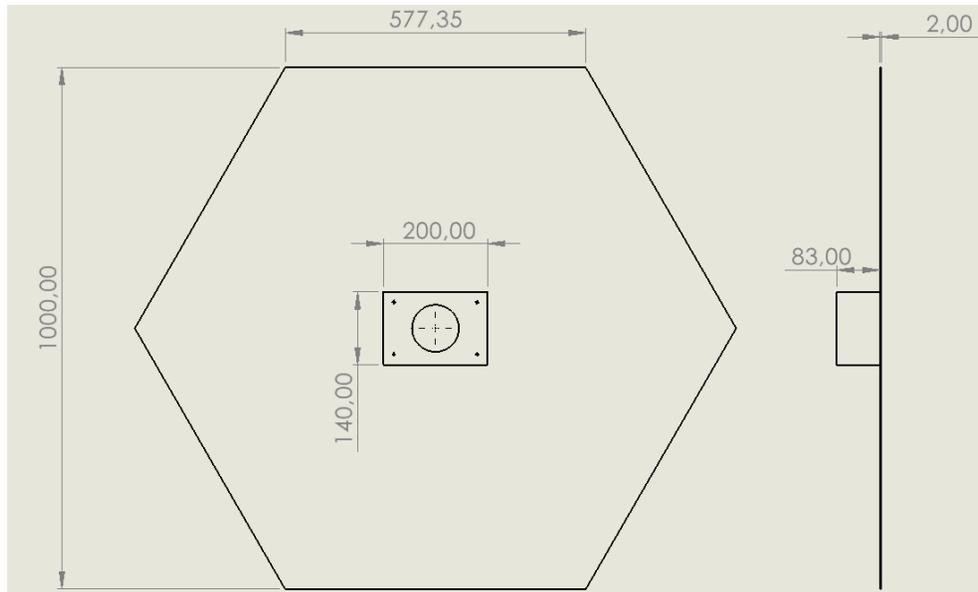
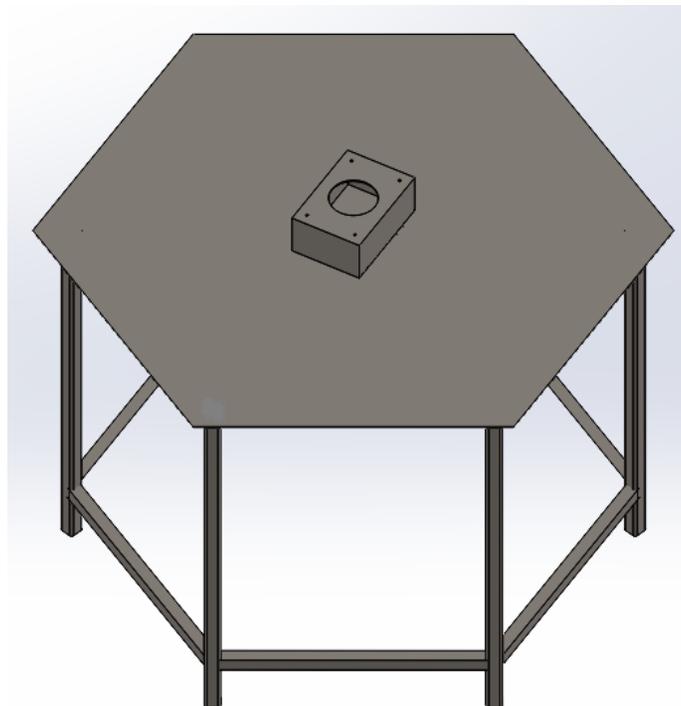
Figura 27*Diseño de la estructura de la mesa.*

Figura 28

Diseño del tablero de la mesa.

**Figura 29**

Ensamble de la estructura con el tablero de la mesa.



En la etapa de construcción se deben seguir los requerimientos físicos de calidad establecidos en la norma NTE INEN 1 649:95.

Análisis Estructural

Se deben realizar pruebas estructurales, las cuales se establecen en la norma NTE INEN 1 648:95, mediante el método de elementos finitos con el uso del software Solidworks.

Para el análisis de la mesa de trabajo se utilizaron los siguientes materiales:

- Material ASTM A36 Acero
- Perfil cuadrado de 30 x 30 x 2 mm
- Ángulo de acero de 30 x 30 x 3 mm

En la Tabla 23 se detallan las características mecánicas del material:

Tabla 23

Propiedades mecánicas del acero ASTM A36.

Propiedad	Valor	Unidades
Módulo elástico	2×10^5	N/mm^2
Módulo cortante	$7,93 \times 10^4$	N/mm^2
Densidad	$7,85 \times 10^3$	Kg/m^3
Límite de tracción	400	N/mm^2
Límite elástico	250	N/mm^2

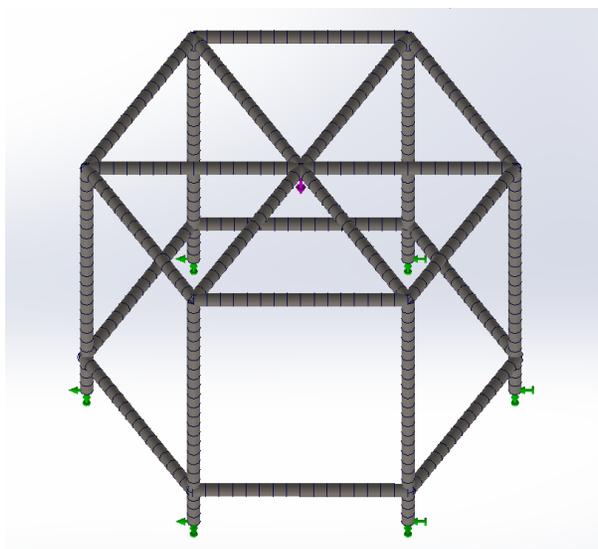
La norma NTE INEN 1 648:95 establece los tipos de ensayos que se deben realizar, los cuales son:

Ensayo de carga estática vertical

Según la norma, la mesa debe ser sometida a una fuerza vertical para el análisis estático en Solidworks, se selecciona el material de la estructura (ASTM A36), en la Figura 30, se muestran los puntos de restricción y la carga estática vertical de 1250 N aplicada en el centro de la mesa, como lo indica la norma técnica.

Figura 30

Distribución de la carga y restricciones en la estructura de la mesa.



En la Figura 31 y la Tabla 24 se detallan los resultados del análisis estático de la estructura de la mesa para la carga vertical. Los puntos con mayor tensión (en color rojo) coinciden con aquellos en donde las deformaciones son mayores y el factor de seguridad tiene su valor mínimo.

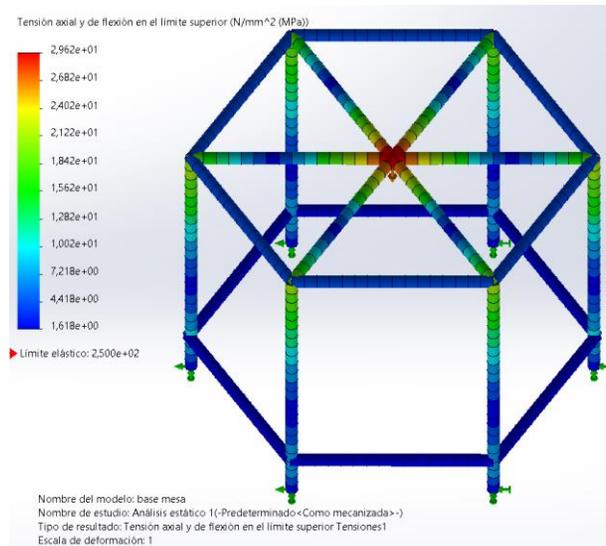
Tabla 24

Análisis estático de la estructura de la mesa para carga vertical.

Propiedad	Valor		Unidades
	Min.	Max.	
Tensión	1,62	29,62	N/mm^2
Deformación unitaria	1×10^{-30}	$5,77 \times 10^{-1}$	mm
Masa	26,7		Kg
Volumen	3.4×10^6		mm^3
Factor de seguridad min	8,4		-

Figura 31

Resultados del análisis estático en la estructura de la mesa para carga vertical.

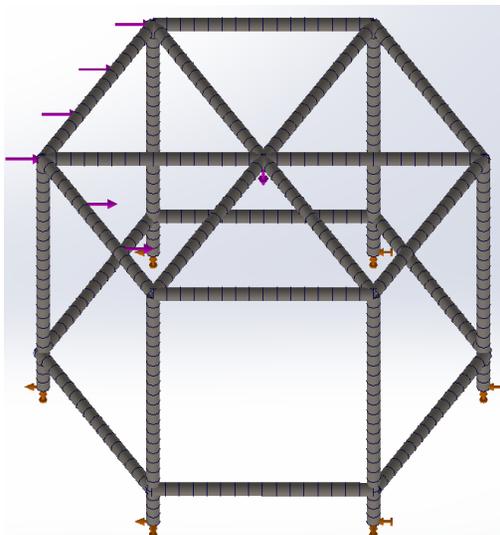


Ensayo de carga estática horizontal

Según la norma se debe fijar una fuerza horizontal de 600 N y una fuerza vertical de 300 N, en la Figura 32, se muestran los puntos de restricción, la carga estática vertical en el centro de la mesa y la carga estática horizontal aplicado en el extremo de la mesa, como lo indica lo norma.

Figura 32

Distribución y restricciones en la estructura de la mesa para carga horizontal.



En la Figura 33 y la Tabla 25 se detallan los resultados del análisis estático de la estructura de la mesa para la carga horizontal. Los puntos con mayor tensión (en color verde) coinciden con aquellos en donde las deformaciones son mayores y el factor de seguridad tiene su valor mínimo.

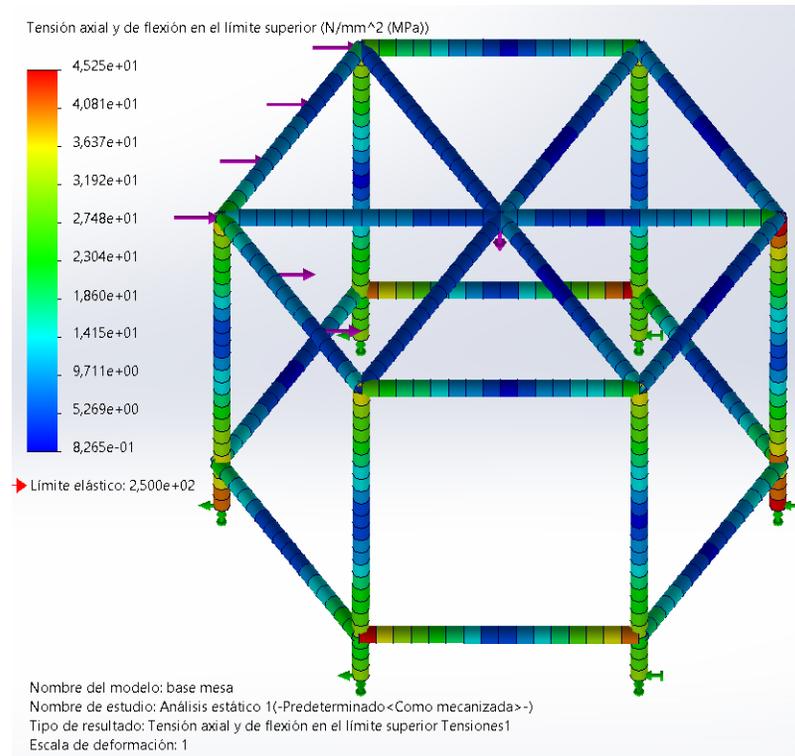
Tabla 25

Resultados del análisis estático en la estructura de la mesa para carga horizontal.

Propiedad	Valor		Unidades
	Min.	Max.	
Tensión	0,83	45,25	N/mm^2
Deformación unitaria	1×10^{-30}	1,27	mm
Masa	26,7		Kg
Volumen	3.4×10^6		mm^3
Factor de seguridad min	5,5		-

Figura 33

Análisis estático en la estructura de la mesa para carga horizontal.



Debido a que la tensión máxima no sobrepasa el límite elástico en ninguno de los dos análisis estructurales realizados, la deformación unitaria máxima es pequeña y considerando que la carga aplicada sobrepasa las condiciones normales de uso que se aplica a la mesa de trabajo, se concluye que el diseño estructural realizado es adecuado.

Diseño de la banda transportadora

Parámetros de diseño

El diseño de la banda transportadora se define de tipo horizontal, de una sola hilera, el material de la estructura de la banda transportadora es de plancha de tool, las dimensiones de la banda en las cuales se va a efectuar el proceso son determinadas por la disponibilidad del área de trabajo del COBOT y del área de trabajo de la mesa; también, el peso máximo de los productos a transportar está limitado por las características del efector final.

La banda transportadora que se utiliza en el empaquetado de productos presenta las siguientes consideraciones:

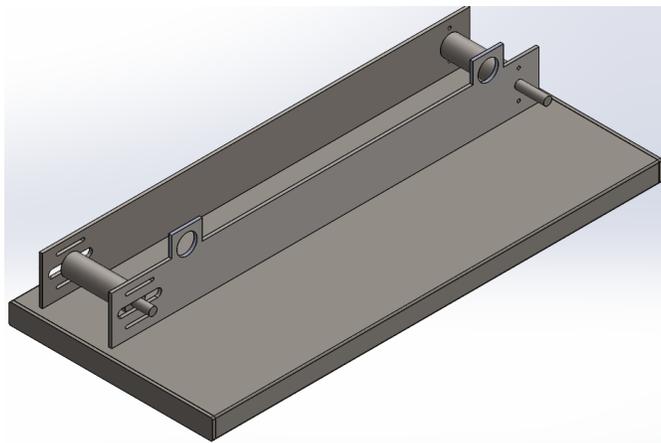
- Longitud útil: 360 mm
- Tipo de recorrido: en línea recta.
- Ancho Útil: 65 mm
- Longitud total: 420 mm
- Ancho total: 71 mm
- Peso máximo del producto: 0,5 kg
- Material: Acero ASTM A36

Diseño CAD

En la Figura 34, se presenta el diseño CAD de la banda transportadora, la banda consta de una base, dos paredes paralelas y dos rodillos que permiten el movimiento a la banda sin fin.

Figura 34

Diseño CAD de la banda transportadora.

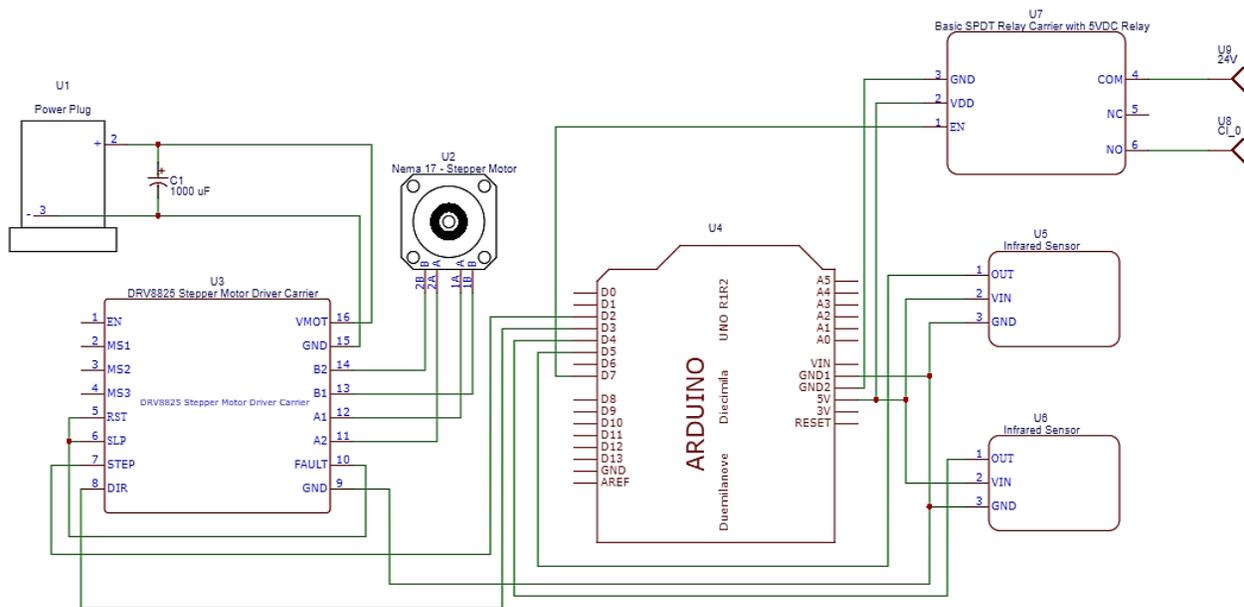


Circuito Electrónico

El circuito electrónico de la banda transportadora consta de los componentes detallados en el esquemático de la Figura 35, para mayor información referirse al Anexo 2.

Figura 35

Esquemático del circuito electrónico



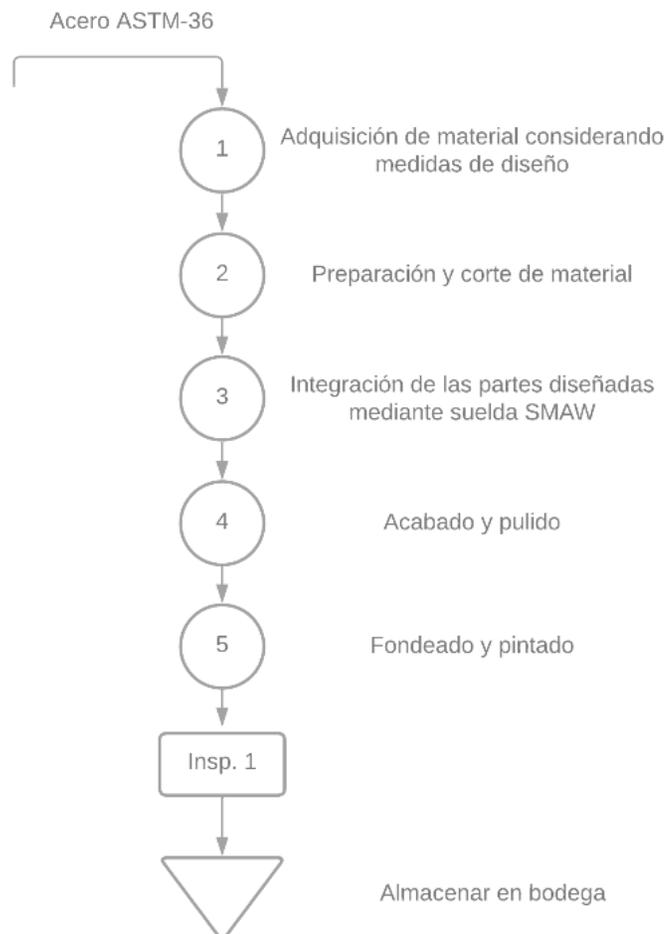
Construcción de la celda robotizada

Construcción de la estructura de la celda

Para la etapa de construcción de la celda se consideran las características de diseño y el proceso a seguir que se evidencia en el diagrama de la Figura 36.

Figura 36

Diagrama de proceso de la construcción estructural de la celda.

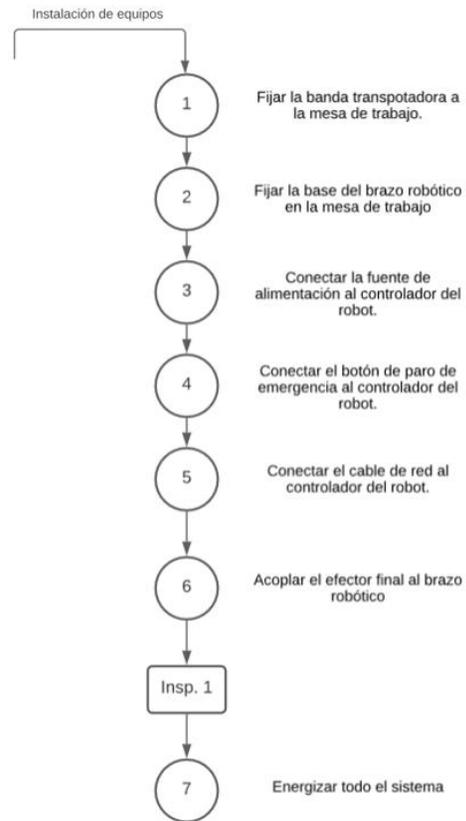


Instalación de accesorios en la mesa de trabajo

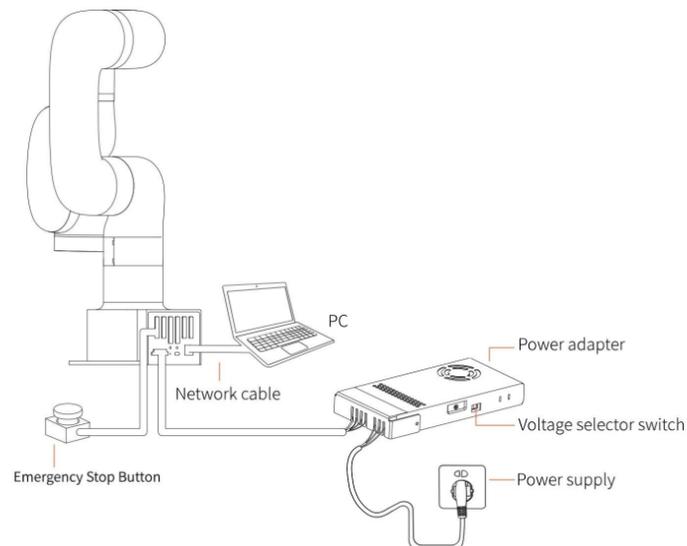
Para la integración de los sistemas mecánicos, eléctricos y electrónicos de la celda se deben seguir el proceso de la Figura 37 y el diagrama de conexión del cobot se muestra en la Figura 38, lo que dan como resultado la estructura mostrada en la Figura 39.

Figura 37

Diagrama de proceso instalación de accesorios en la celda

**Figura 38**

Esquema de conexión del COBOT LITE 6.



Nota. Elementos requeridos para el funcionamiento del COBOT Lite 6, Tomado de (Ufactory, 2021b).

Figura 39

Construcción de la celda robotizada.



Comunicación entre el robot y dispositivos

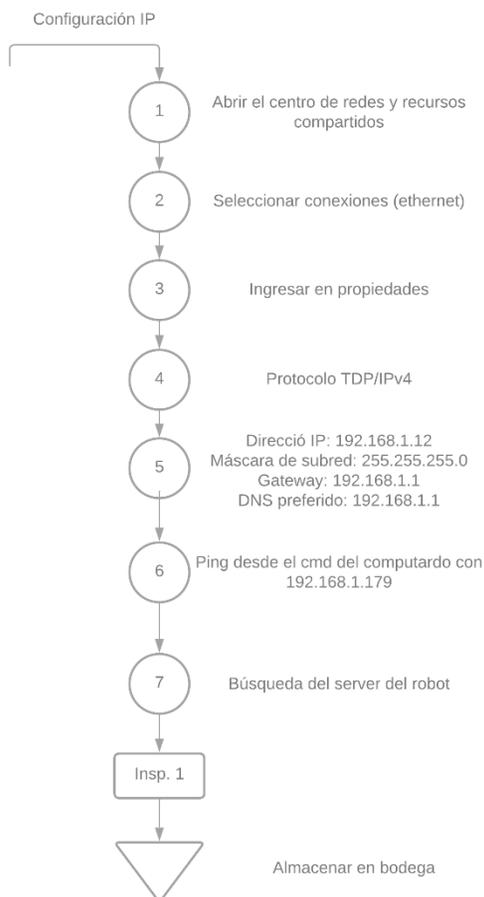
Comunicación PC – COBOT

Configuración IP

Es necesario configurar la dirección IP del ordenador con la caja de control del robot de modo que se encuentren en el mismo segmento de red. La dirección IP de la caja de control viene determinada por el fabricante que es 192.168.1.197. A continuación, se muestra el diagrama de proceso (Figura 40) para la configuración de la IP en el sistema operativo de Windows.

Figura 40

Diagrama de proceso para configuración IP del robot y ordenador.

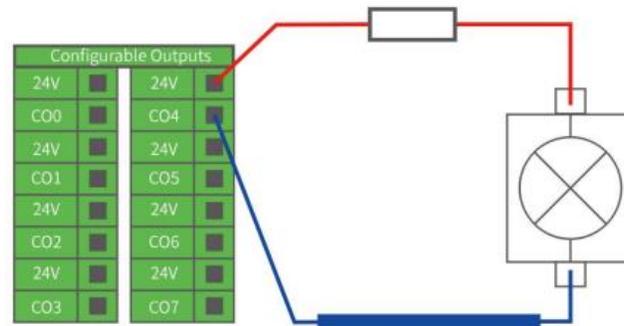


Configuración de la salida digital del controlador

En la Figura 41, se muestra cómo conectar a una salida digital con una carga resistiva, según el Anexo 1 (Manual se usuario del cobot); para una carga resistiva es necesario conectar una resistencia, a más de un fusible como elemento de protección. Se utiliza una luz piloto de color rojo como indicador para cuándo la ventosa no succiona el producto en la aplicación de empaquetado, en la Tabla 26 se especifican las características de funcionamiento de la luz piloto.

Figura 41

Diagrama de conexión de una carga resistiva al controlador.



Nota. Se requiere de una resistencia y un fusible para protección. Tomado de (Ufactory, 2021b).

Tabla 26

Parámetros de funcionamiento.

Propiedad	Valor	Unidades
Voltaje de funcionamiento (AC/DC)	24	V
Corriente	20	mA
Diámetro de orificio de montaje	16	mm
Color	Rojo	-

Para la implementación de la carga resistiva en el controlador se debe calcular mediante la ley de Ohm el valor de la resistencia y su potencia, con Ecuación 1 y Ecuación 2 respectivamente:

$$R = \frac{V}{I} \quad \text{Ecuación 1}$$

$$P = V * I \quad \text{Ecuación 2}$$

Reemplazando el valor del voltaje e intensidad nominal a la que trabaja la luz piloto en la Ecuación 1 y Ecuación 2, se determina que el valor de la resistencia es de $R = 1,2 \text{ K}\Omega$ y una potencia de $P = 0,48 \text{ watts}$; además, la corriente de funcionamiento del fusible es del 10% sobre el valor de la intensidad nominal de funcionamiento de la luz piloto.

Comunicación PC - Banda

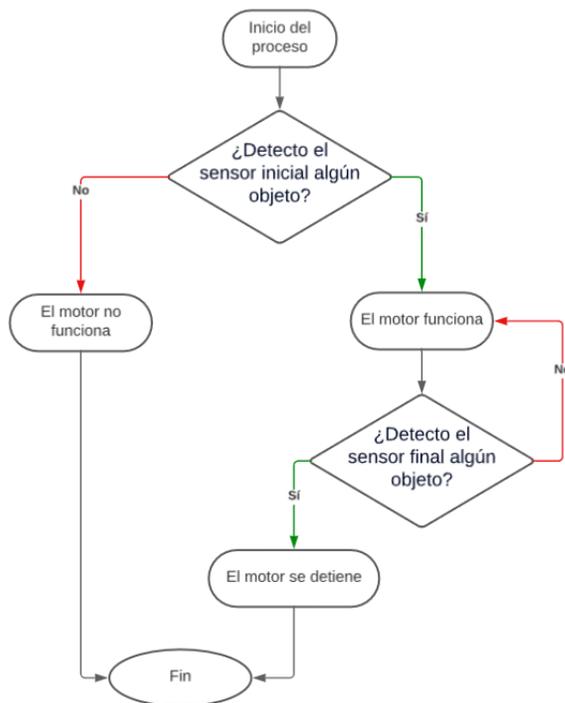
Para la comunicación de la banda se utilizó una tarjeta de adquisición de datos (Arduino Uno), con la cual se realiza la automatización de la banda transportadora; los elementos que interactúan en este sistema son:

- Dos sensores infrarrojos
- Un motor nema 17 paso a paso con caja reductora.

En la Figura 42, se muestra el diagrama de flujo del funcionamiento de la banda transportadora.

Figura 42

Diagrama de flujo del funcionamiento de la banda.

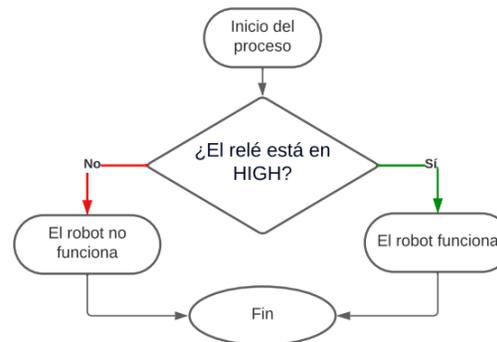


Comunicación COBOT - Banda

Para la comunicación del COBOT con la banda se conecta la salida del relé a la entrada digital de la caja de control. En la Figura 43, se detalla la lógica de funcionamiento de la entrada digital del controlador.

Figura 43

Diagrama de flujo del funcionamiento de la banda.



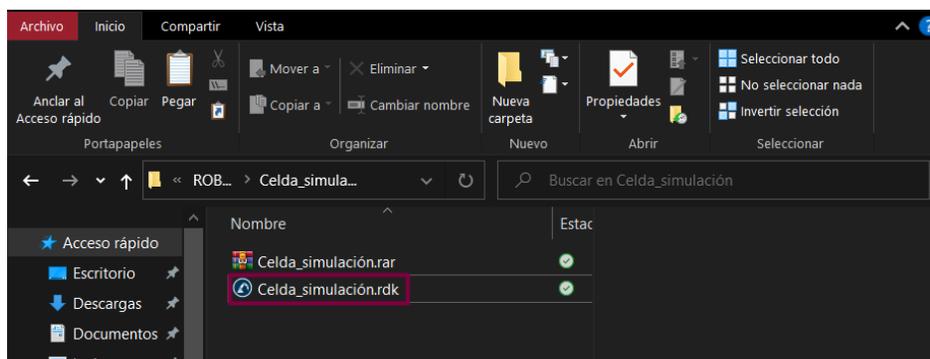
Programación offline en el software seleccionado

Para comenzar con el proceso de la programación offline para la aplicación de empaquetado de productos se utiliza el software RoboDK, este software permite definir las trayectorias de manera virtual y su correspondiente simulación en el layout creado.

En primera instancia se debe acceder al layout desarrollado de la celda robotizada en RoboDK, ingresando desde la carpeta contenedora se debe ejecutar el archivo, como se ejemplifica en la Figura 44.

Figura 44

Ejecución del Layout de la celda robotizada.

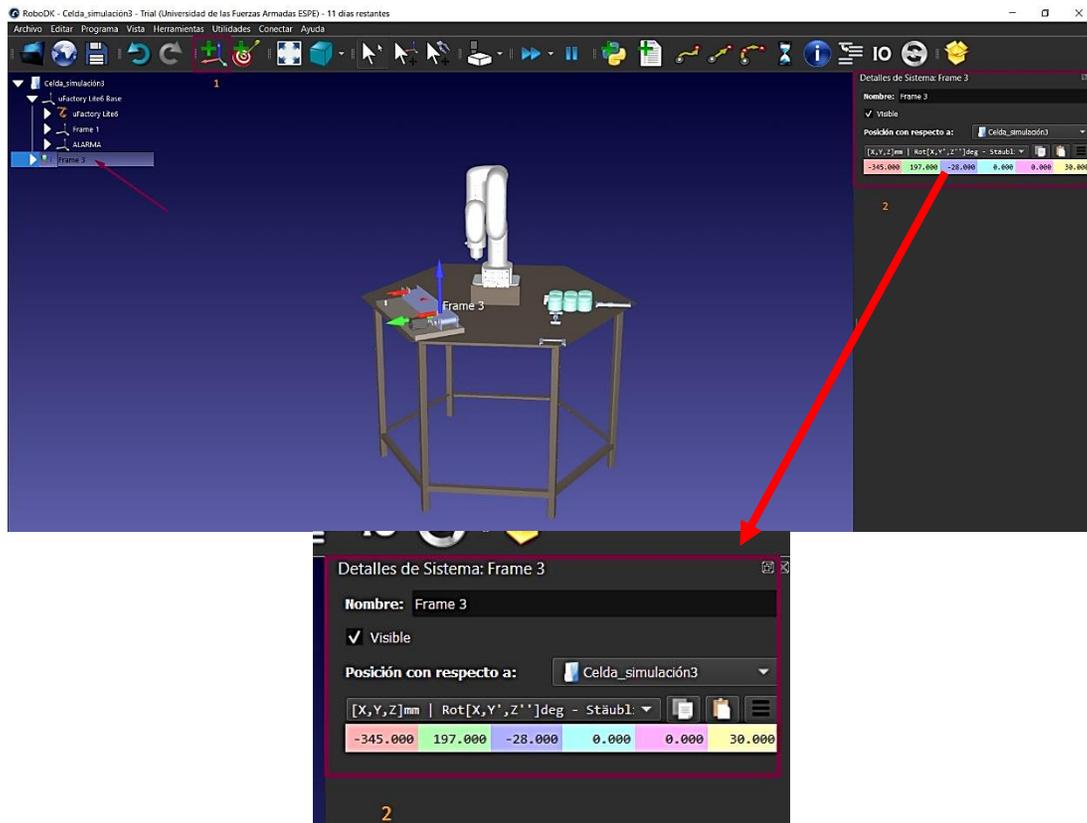


Tras la ejecución del archivo denominado *Celda_simulación*, se debe crear un Frame y definir la configuración necesaria del punto cero de este nuevo sistema de coordenadas creado, el cual marca el punto de llegada del producto al final del trayecto de la banda, a más

de generar la señal que activa el movimiento del robot. Las coordenadas del *Frame* generado se muestran en la Figura 45.

Figura 45

Creación de Frame y ajuste de posición.



La siguiente actividad es determinar los puntos que conforman la trayectoria planificada, para ello se debe seleccionar el robot, en lado izquierdo de la pantalla del layout creado (1), a continuación, mover al robot a la posición deseada (2), para luego crear un *Target* (3), configurar el nombre a conveniencia del usuario dando un clic derecho sobre el *Target* creado (4); y así sucesivamente con los puntos necesarios para generar una trayectoria.

El este procedimiento para crear un target se visualiza en la Figura 46 y en la Figura 47 se muestra el proceso para renombrar un *Target*.

Figura 46

Creación y configuración de los puntos de la trayectoria.

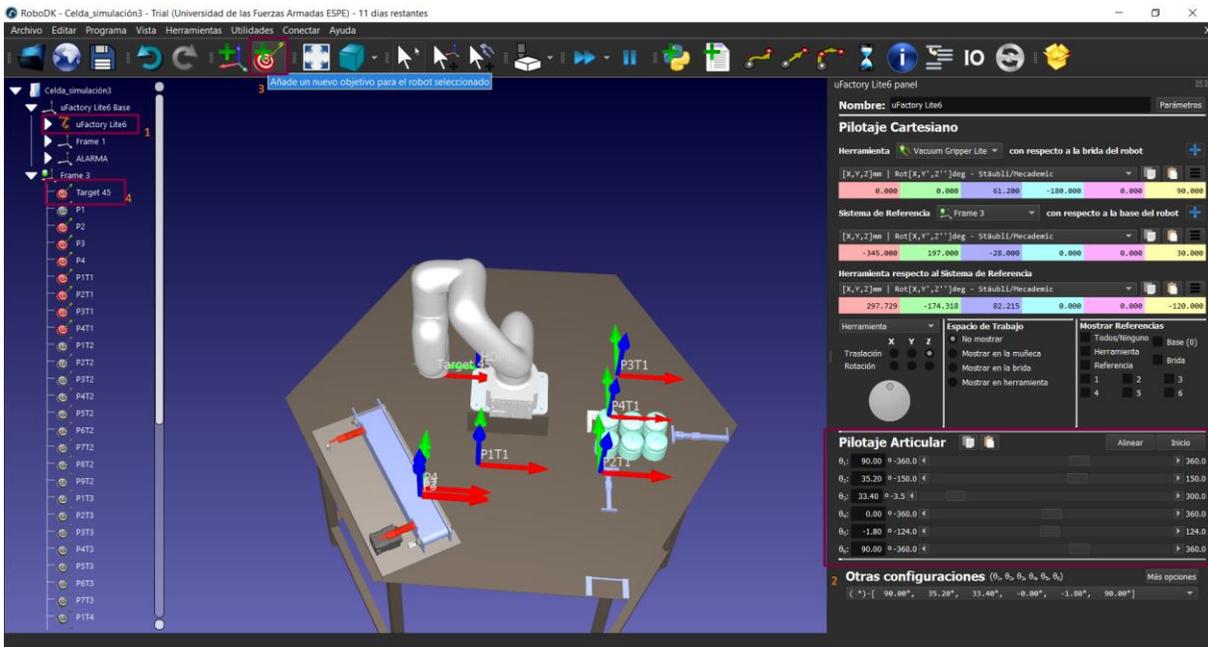
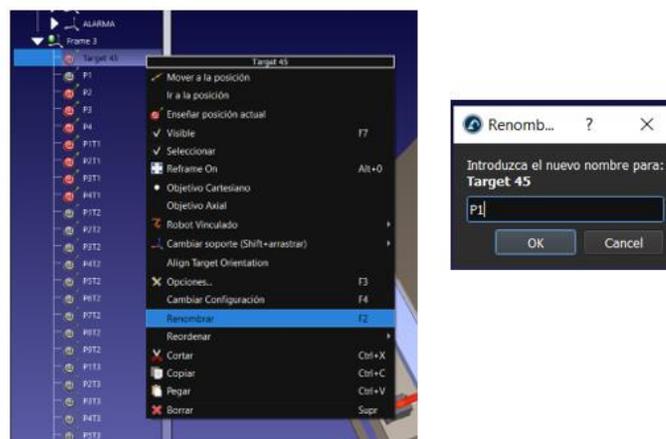


Figura 47

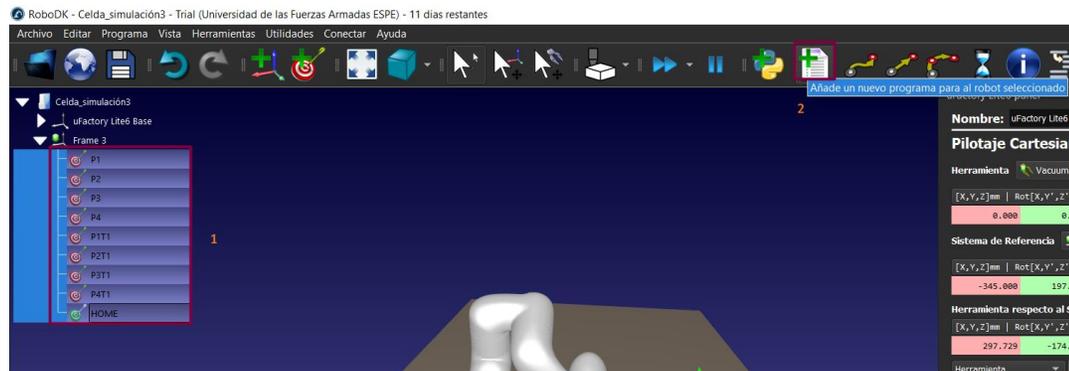
Configuración de nombre del punto generado.



Una vez creados los puntos se genera un programa el cual contiene la trayectoria deseada, para ello se seleccionan los puntos creados (1) y se da clic en el ícono nombrado *Añade un nuevo programa para el robot seleccionado* (2), como se muestra en la Figura 48.

Figura 48

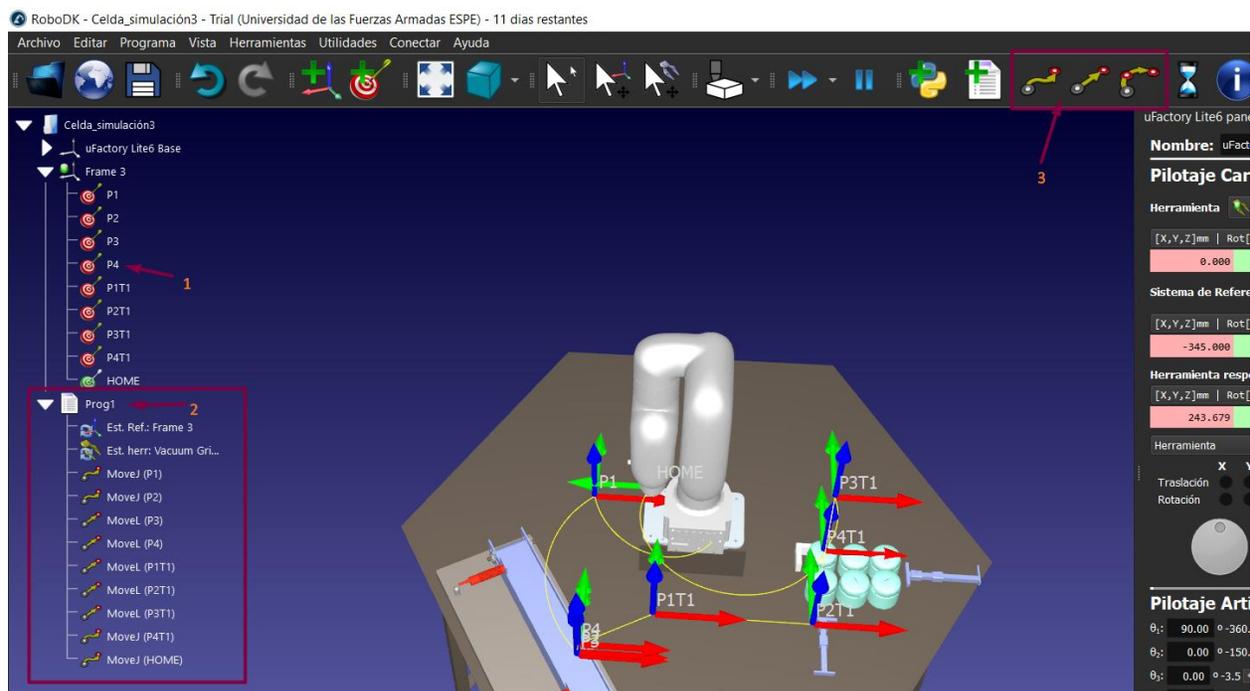
Creación del programa para generar la trayectoria.



Una vez creado el programa el usuario debe crear los movimientos necesarios para concluir con la trayectoria, seleccionando el punto (1), programa creado (2) y el tipo de movimiento que desea realizar (3); el software RoboDK cuenta con tres tipos de movimiento: articular, lineal y circular; el procedimiento descrito se evidencia en la Figura 49.

Figura 49

Creación de movimientos necesarios para completar una trayectoria.

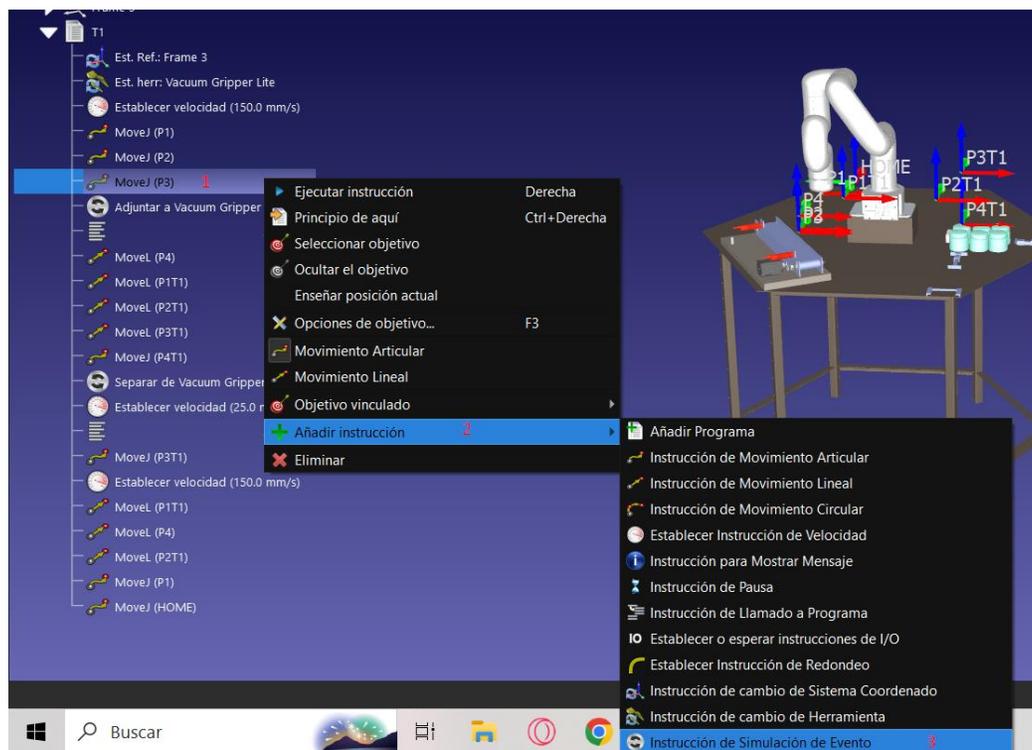


La trayectoria creada debe ser acompañada de eventos, que ayudan a visualizar la simulación de la aplicación a ser ejecutada en la vida real; para ello se definen eventos como: el de adjuntar y dejar los objetos a la ventosa, tiempos de espera, establecimiento de velocidad y aceleración, con los que se producen los movimientos.

Para generar un evento se debe dar clic derecho sobre el punto anterior al que se ejecuta dicha acción (1), seleccionar la opción *Añadir instrucción* (2), y después la opción *Instrucción de Simulación de Evento* (3), esta secuencia de instrucciones se observa en la Figura 50.

Figura 50

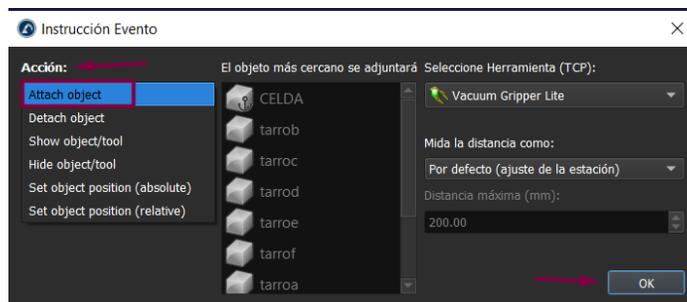
Secuencia de instrucciones para crear una "Instrucción de evento".



En la nueva ventana emergente se debe realizar la configuración mostrada en la Figura 51 y de la lista de acciones seleccionar *Attach Object* para simular el encendido y succión de la ventosa; para el caso del apagado y expulsión de la ventosa se selecciona la acción denominada *Detach Object*.

Figura 51

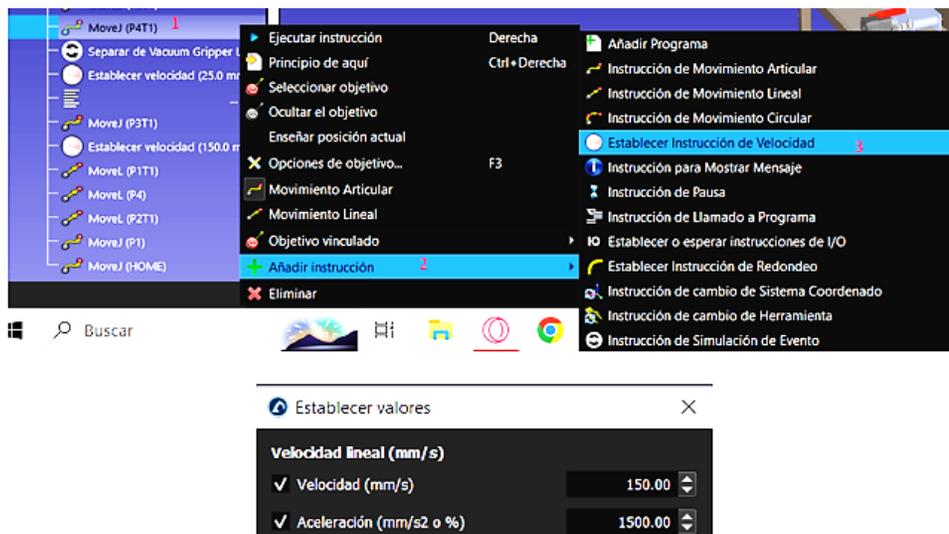
Generación del evento de encendido y succión de la ventosa.



En el caso de configurar la velocidad y aceleración del movimiento, se selecciona la opción *Establecer instrucción de velocidad* y se establecen los valores, de la Figura 52, cabe mencionar que los valores máximos de velocidad y aceleración del robot son de 500 mm/s y 2000 mm/s² respectivamente.

Figura 52

Creación de la instrucción de velocidad y configuración de parámetros.



Para activar la ventosa del robot en la vida real se debe crear una instrucción denominada *Instrucción de Llamado a Programa* (Figura 53), el cual permite insertar el código de activación para la ventosa y lectura de la entrada digital del robot (Figura 54), que es reconocido al momento de cargar el archivo final en el controlador del COBOT.

Figura 53

Creación de la instrucción de Llamado a Programa.

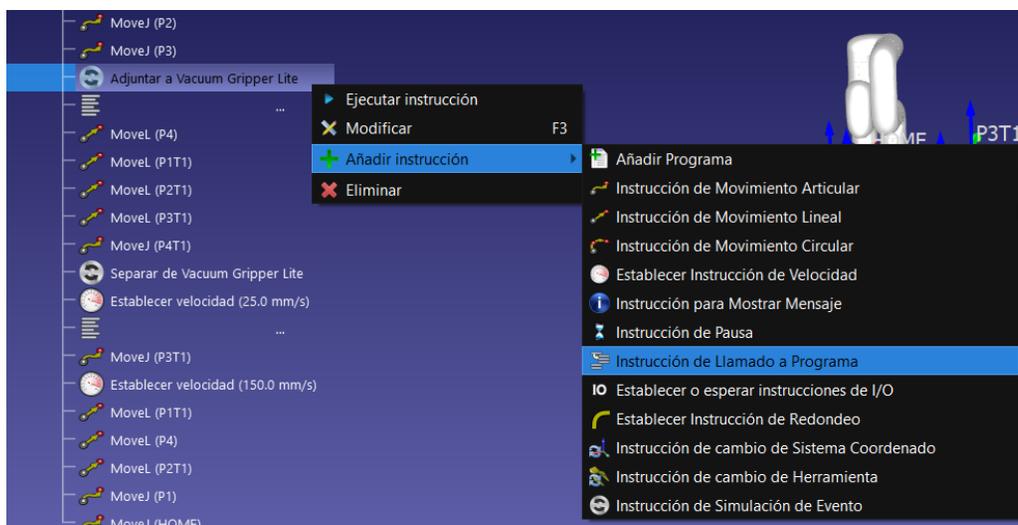
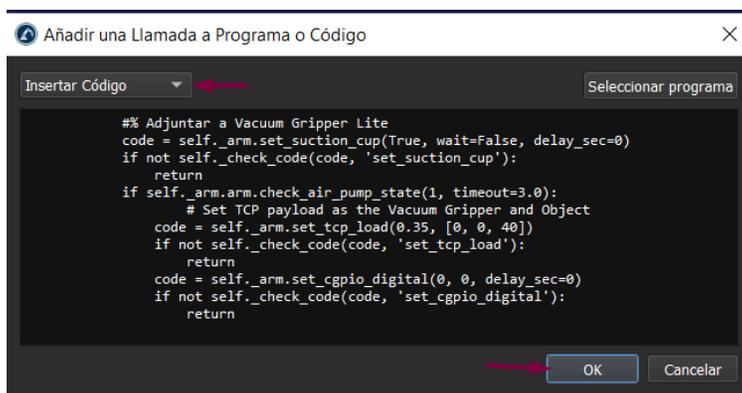


Figura 54

Entrada digital del COBOT y código de activación de la ventosa.



La Figura 55 muestra la programación completa de una trayectoria con movimientos e instrucciones necesarias para llevar a cabo la simulación y considerando las acciones para la ejecución en la vida real.

Figura 55

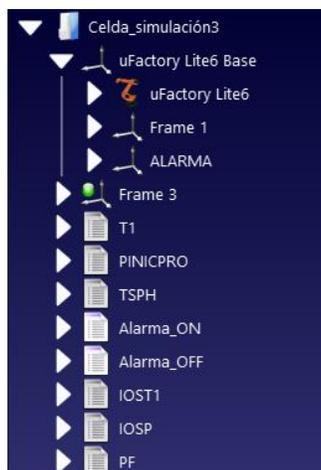
Definición de movimientos e instrucciones para la generación de trayectorias.



A continuación, para completar el programa de la trayectoria, es necesario crear programas con las condiciones requeridas para su adecuada simulación (Figura 56), que en este caso son:

Figura 56

Condiciones necesarias para la creación de la trayectoria.



1. PINICPRO: Programa para el posicionamiento del robot en el punto inicial de toma del producto (Figura 57).

Figura 57

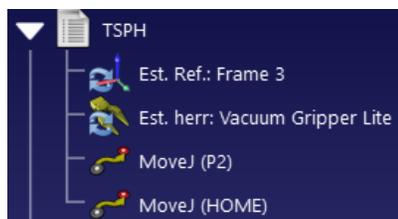
Instrucciones del programa PINICPRO.



2. TSPH: Programa que se ejecuta en caso de no existir un producto en el punto inicial o en caso de que la ventosa no succione de manera correcta el producto, llevando al robot a la posición de HOME (Figura 58).

Figura 58

Instrucciones del programa TSPH.



3. Alarma_ON: Programa condicionado para encender la alarma visual (luz piloto roja) en caso de no ser tomado el producto por la ventosa del robot (Figura 59).

Figura 59

Programación del programa Alarma_ON



4. Alarma_OFF: Programa condicionado para apagar la alarma visual (luz piloto roja) en caso de ser tomado el producto por la ventosa del robot (Figura 60).

Figura 60

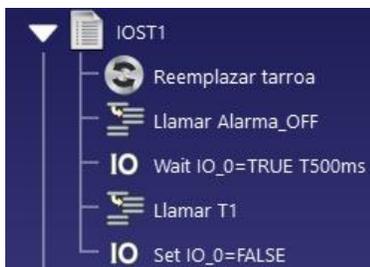
Codificación del programa Alarma_OFF.



5. IOST1: Programa creado para generar la trayectoria con las condiciones de: reemplazo del producto para ejecutar varias veces la trayectoria, alarma apagada, reconocimiento de la entrada digital, la ejecución del programa de la trayectoria que en este caso se denomina *T1*. Este programa se ejecuta cuando el producto es reconocido por la ventosa. (Figura 61).

Figura 61

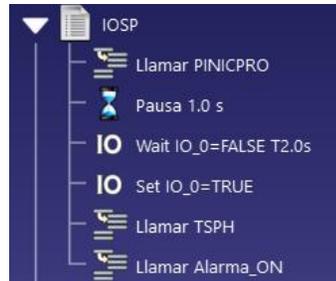
Instrucciones del programa IOST1.



6. IOSP: Programa creado para generar la trayectoria con las condiciones de: llamado del programa PINICPRO, una pausa de 1s, una espera de 2s en la entrada digital, el estado de la salida digital como verdadero, llamado del programa TSPH y llamado del programa Alarma_ON. Este programa es ejecutado cuando la ventosa no succiona el producto de manera adecuada o no existe el producto en el punto de partida (Figura 62).

Figura 62

Codificación del programa IOSP.



7. PF: Programa principal en el que se ejecuta la simulación de la trayectoria con todos los requerimientos antes descritos (Figura 63).

Figura 63

Instrucciones del programa PF.



Tras realizar el programa final, lo siguiente es ejecutarlo y visualizar la simulación, con las condiciones programadas. En la Figura 64 se presenta la simulación de la trayectoria al ser tomado el producto, mostrando el estado de la alarma visual; y en la Figura 65 se presenta la simulación de la trayectoria al no ser reconocido el producto en el punto de partida.

Figura 64

Collage de imágenes de la ejecución de la trayectoria.

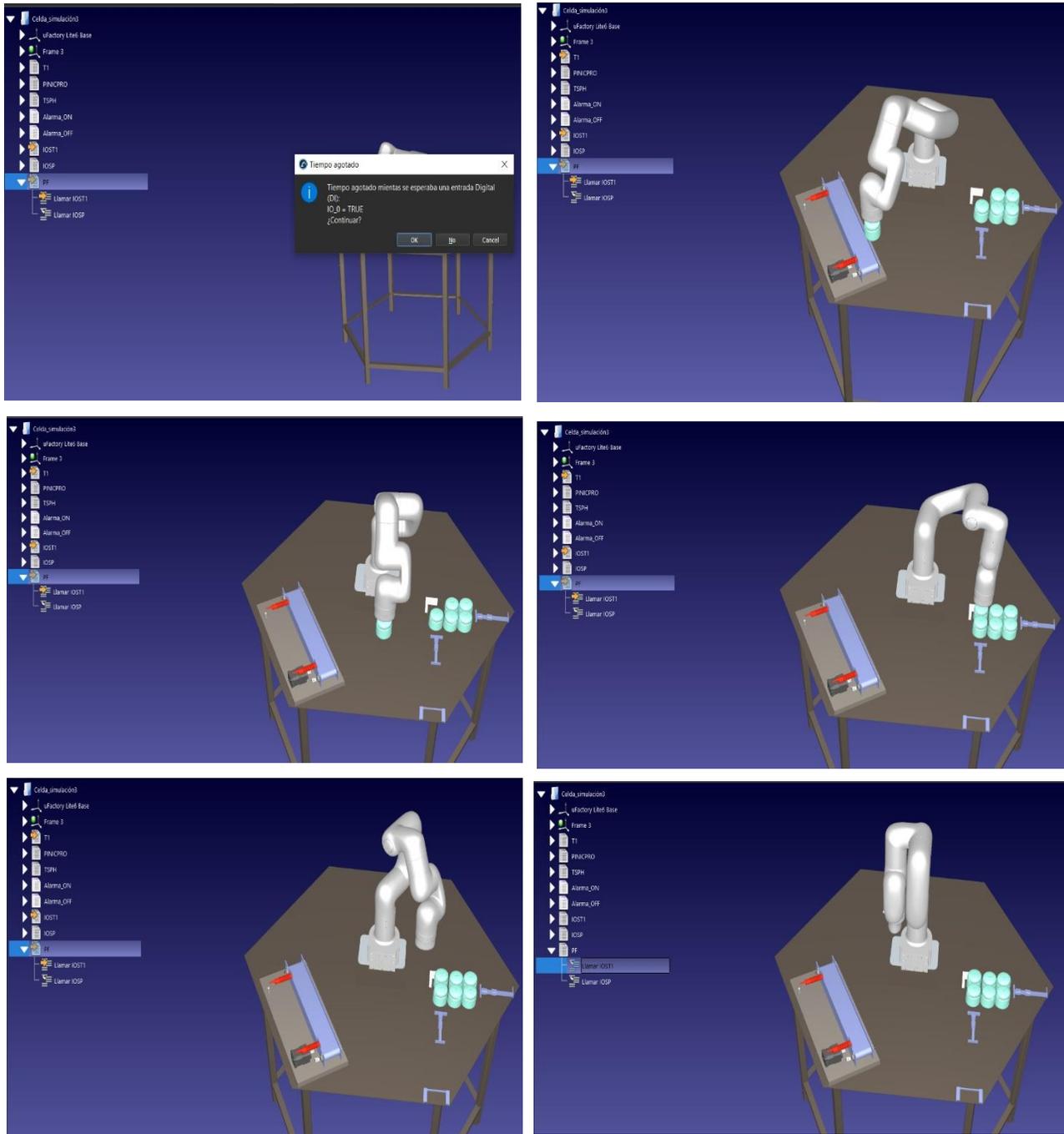
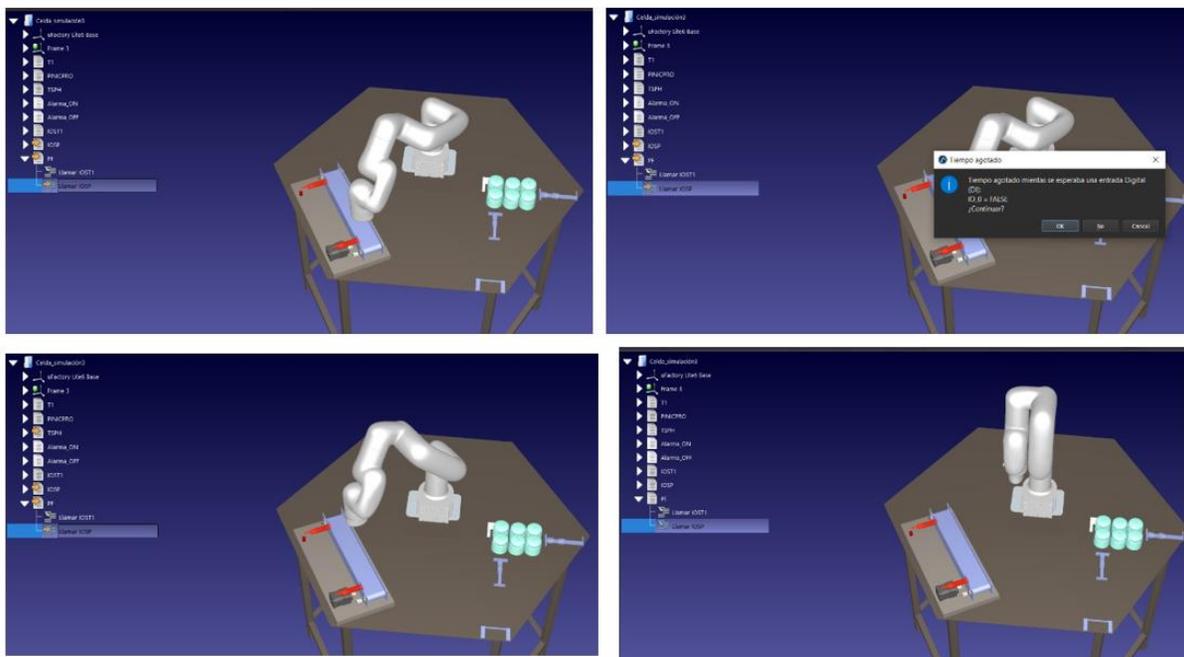


Figura 65

Collage de imágenes de la segunda trayectoria.



Una vez terminada la simulación y creación de la trayectoria se procede a generar el programa para continuar con el proceso de programación offline, dando clic derecho sobre el programa final y seleccionando la opción *Seleccionar Post-procesador* (Figura 66), luego en la ventana emergente se selecciona el post-procesador denominado como *Lite6* (Figura 67).

Figura 66

Selección de la opción Post-procesador.

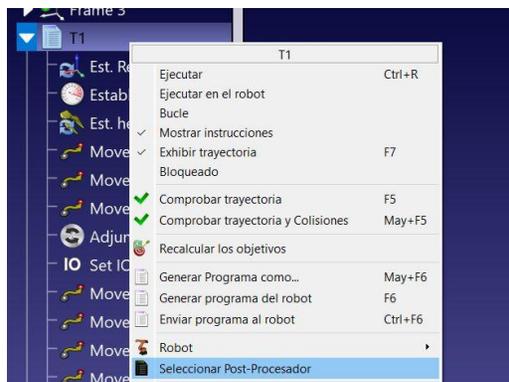
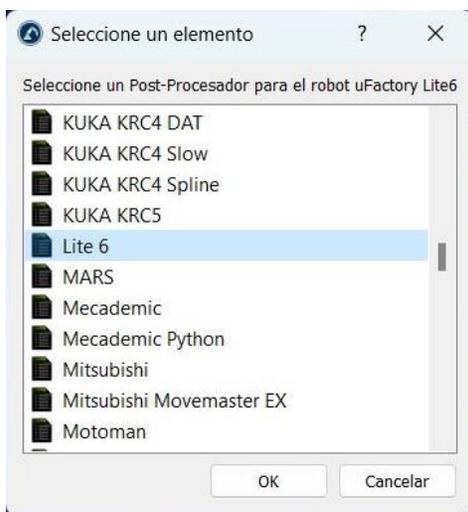


Figura 67

Selección del post-procesador denominado Lite6.



A continuación, se procede a generar el código, dando clic derecho sobre el programa y seleccionando la opción *Generar programa del robot* (Figura 68), finalmente se presenta el código post-procesado en lenguaje Python (Figura 69).

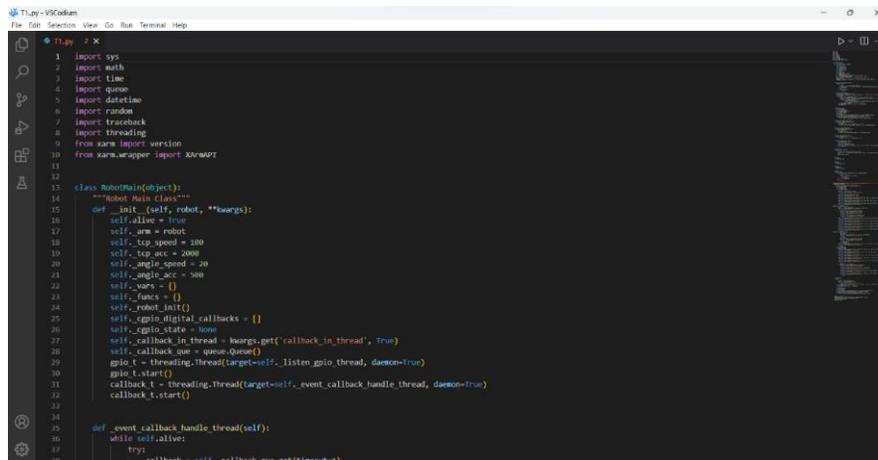
Figura 68

Selección de la opción "Generar programa del robot".



Figura 69

Código en lenguaje Python post procesado desde RoboDK.



```

1 import sys
2 import math
3 import time
4 import queue
5 import datetime
6 import random
7 import traceback
8 import threading
9 from xarray import version
10 from xarray.wrapper import XArrayDT
11
12
13 class RobotMain(object):
14     """Robot Main Class"""
15     def __init__(self, robot, **kwargs):
16         self.alive = True
17         self.ara = robot
18         self.tsp_speed = 100
19         self.tsp_acc = 2000
20         self.angle_speed = 20
21         self.angle_acc = 500
22         self.kwargs = {}
23         self.functs = {}
24         self._robot_init()
25         self._gpio.digital_callbacks = []
26         self._gpio.state = None
27         self._callback_in_thread = kwargs.get('callback_in_thread', True)
28         self._callback_queue = queue.Queue()
29         gpio.t = threading.Thread(target=self._listen_gpio_thread, daemon=True)
30         gpio.t.start()
31         callback_t = threading.Thread(target=self._event_callback_handle_thread, daemon=True)
32         callback_t.start()
33
34
35     def _event_callback_handle_thread(self):
36         while self.alive:
37             try:
38                 callback = self._callback_queue.get(timeout=1)

```

Tras obtener el código de Python se procede a generar el archivo *T1.py* de la Figura 70 el cual se obtiene dando clic en la pestaña *File* seguido de la opción *Save as* y finalmente en la ventana emergente se da clic en el botón *Guardar* obteniendo así el archivo *.py*, la secuencia de pasos a seguir se encuentra en la Figura 71.

Figura 70

Archivo *.py*

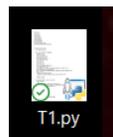
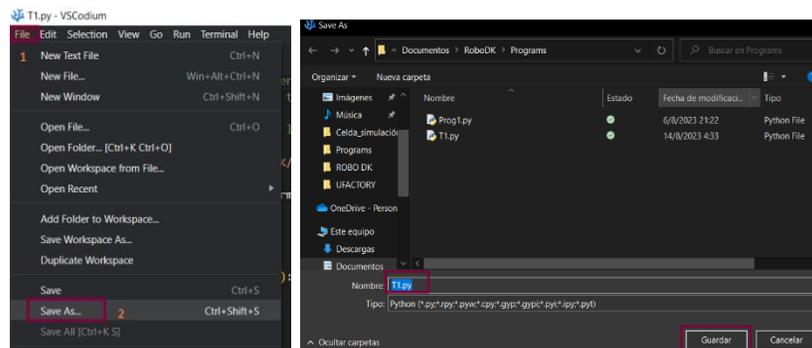


Figura 71

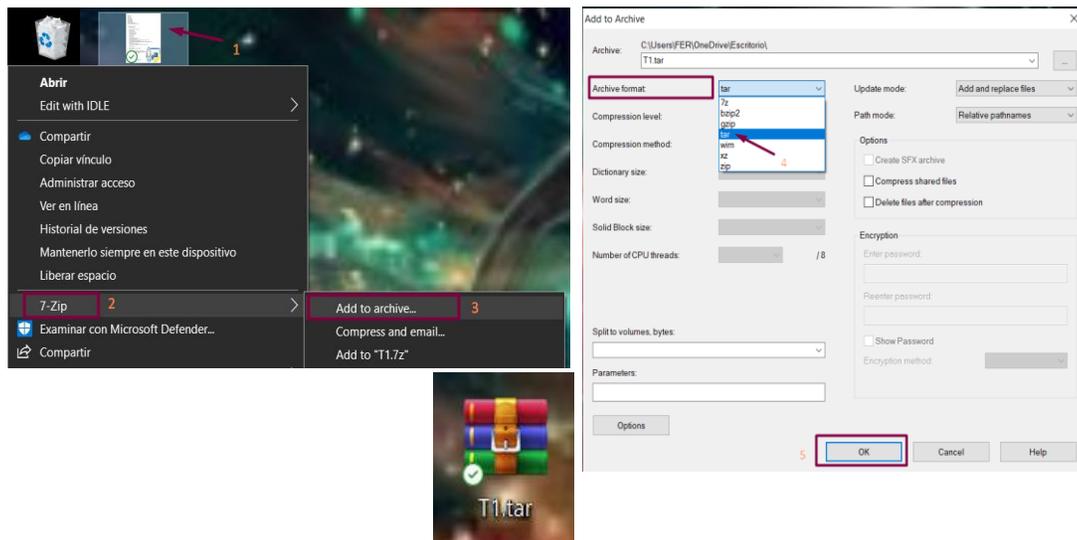
Procedimiento para obtener el archivo *.py*



Luego de obtener el archivo del código Python se procede a comprimirlo, para ello se debe tener previamente instalado el programa 7-Zip en el ordenador; el primer paso es dar clic derecho sobre el archivo *T1.py* (1) y en las ventanas emergentes se selecciona la opción 7-Zip (2), a continuación, la opción *Add to archive* (3); en la nueva ventana en el apartado *Archive format*, se selecciona la extensión *tar* (4) y luego se da clic en *OK* (5). De este modo se obtiene un archivo llamado *T1.tar*, este proceso se ejemplifica en la Figura 72.

Figura 72

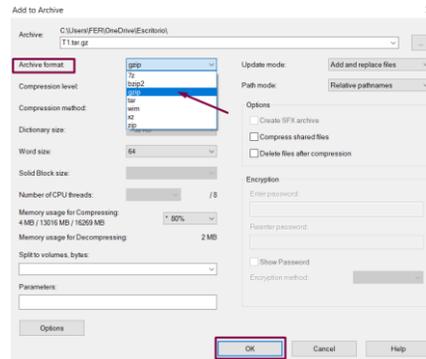
Creación del archivo con extensión ".tar".



Siguiendo con el proceso, se debe comprimir una vez más el archivo *T1.tar*, con la misma secuencia de la Figura 72, pero en esta ocasión se aplica al archivo *T1.tar* y en la ventana emergente en el apartado *Archive format* se escoge la opción *gzip* (Figura 73) y por último, se da clic en *OK*, teniendo como resultado final un archivo con la extensión *.tar.gz* (Figura 74).

Figura 73

Selección de la extensión .gzip para generar un archivo .gz

**Figura 74**

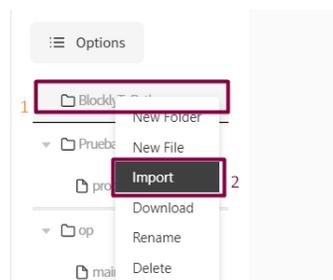
Archivo T1.tar.gz generado.



El siguiente paso es cargar el archivo *T1.tar.gz* en el controlador del robot Xarm Lite 6 para ello se debe seguir la siguiente secuencia expuesta en la Figura 75, primero se escoge la carpeta en la que se almacena el archivo, que en este caso es conocida como *BlockytoPython* y con un clic derecho se escoge la opción *Import*.

Figura 75

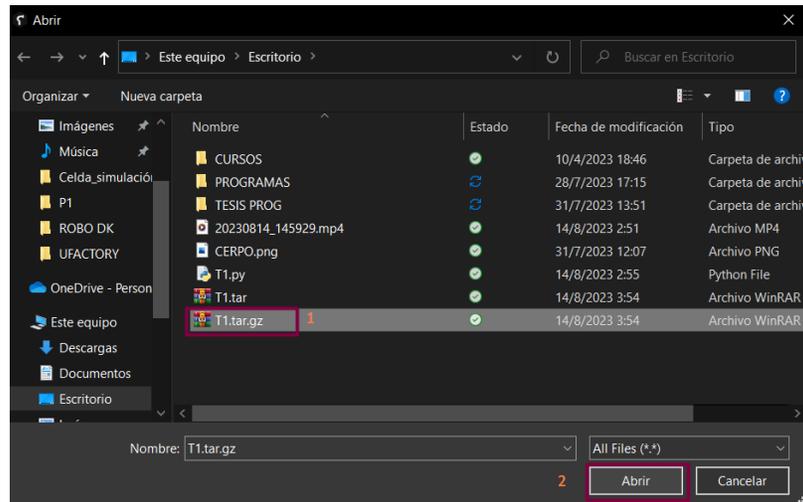
Selección de carpeta y opción de importación.



El siguiente paso es escoger el archivo creado y dar clic en el botón *Abrir*, para cargar el código (Figura 76).

Figura 76

Selección del archivo a cargarse en el controlador del robot.



Finalmente se visualiza el archivo cargado y se procede a ejecutarlo, se aceptan los mensajes de confirmación que aparecen (Figura 77), se evidencia la ejecución del programa creado en el terminal del IDE de Python del software UFACTORY Studio (Figura 78).

Figura 77

Carga del archivo T1.tar.gz, ejecución del mismo y aceptación de mensajes.

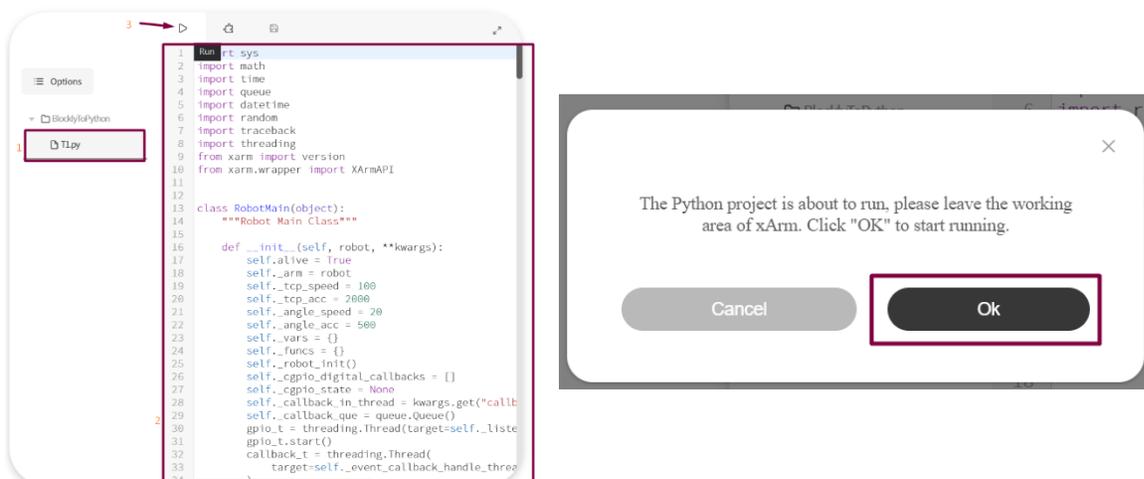


Figura 78

Ejecución del programa de la trayectoria creada en RoboDK.

The figure consists of two side-by-side screenshots of a code editor window. The left screenshot shows the source code of a Python program. The right screenshot shows the same code editor with the execution output displayed in a terminal window at the bottom.

Left Screenshot (Code):

```

1 import sys
2 import math
3 import time
4 import queue
5 import datetime
6 import random
7 import traceback
8 import threading
9 from xarm import version
10 from xarm.wrapper import XArmAPI
11
12
13 class RobotMain(object):
14     """Robot Main Class"""
15
16     def __init__(self, robot, **kwargs):
17         self.alive = True
18         self.arm = robot
19         self.top_speed = 100
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

Right Screenshot (Code and Output):

```

1 import sys
2 import math
3 import time
4 import queue
5 import datetime
6 import random
7 import traceback
8 import threading
9 from xarm import version
10 from xarm.wrapper import XArmAPI
11
12
13 class RobotMain(object):
14     """Robot Main Class"""
15
16     def __init__(self, robot, **kwargs):
17         self.alive = True
18         self.arm = robot
19         self.top_speed = 100
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

Terminal Output (Left Screenshot):

```

~/BlocklyToPython/TLpy
SDK VERSION: 1.13.8
[2023-04-05 04:47:13][399] xArm-Python-SDK Version:1.13.8
ROBOT_IP: 192.168.1.179, VERSION: v2.1.0, PROTOCOL: V1,
DETAIL: 6,9,LL1006,DL1000,v2.1.0, TYPE1300: [0, 0]
change protocol identifier to 3
[clean_error], xArm is ready to move

```

Terminal Output (Right Screenshot):

```

~/BlocklyToPython/TLpy
SDK VERSION: 1.13.8
[2023-04-05 04:47:13][399] xArm-Python-SDK Version:1.13.8
ROBOT_IP: 192.168.1.179, VERSION: v2.1.0, PROTOCOL: V1,
DETAIL: 6,9,LL1006,DL1000,v2.1.0, TYPE1300: [0, 0]
change protocol identifier to 3
[clean_error], xArm is ready to move
[2023-04-05 04:48:00][120] counter val: 1

```

Capítulo V

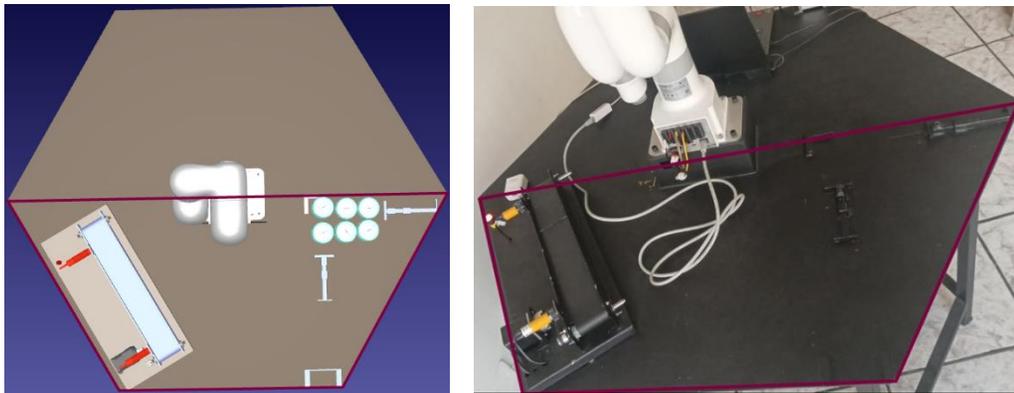
Pruebas y Resultados

Parámetros de funcionamiento de la aplicación de empaquetado del proyecto

Para la aplicación de empaquetado de productos se contó con un área de trabajo encerrada en una semicircunferencia de radio 440mm, que en este caso se encuentra en la parte posterior del COBOT, como se muestra en la Figura 79, lugar donde se realizan pruebas del funcionamiento del robot.

Figura 79

Área de trabajo para la aplicación de empaquetado.



La aplicación de empaquetado fue enfocada para realizar el almacenamiento de 6 productos de diversos tamaños y formas, en sus respectivas cajas.

Operación del brazo robótico y ventosa

El efector final cuenta con un juego de tres ventosas de diferentes tamaños una grande (3 cm), mediana (1.5 cm) y pequeña (1cm) como se muestra en la Figura 80. Se visualiza como el COBOT levanta un producto de 113g, utilizado en la aplicación de empaquetado en la Figura 81. Cabe recalcar que el sistema de encendido del efector final es eléctrico.

Figura 80

Juego de ventosas del brazo robótico.

**Figura 81**

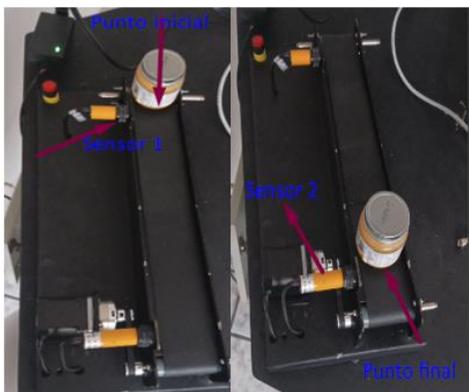
Operación del brazo robótico con una carga de 113 g.

**Operación de la banda transportadora**

La Figura 82 evidencia el funcionamiento de la banda transportadora, la cual está configurada con una velocidad constante de 50 mm/s adecuada para el transporte de los productos seleccionados.

Figura 82

Funcionamiento de la banda.



Prueba de comunicación entre el robot y dispositivos

El objetivo primordial de esta prueba es constatar que los accesorios que conforman la celda robotizada actúen en forma sincrónica para realizar de manera adecuada el control de la trayectoria para el empaquetado de un producto.

Se debe energizar los sistemas que conforman la celda robotizada, se inicia el proceso ejecutando el programa de la trayectoria en el software UFACTORY Studio, previamente creada en RoboDK como se presentó previamente en las Figura 49 y Figura 78 respectivamente.

Ejecutado el programa, en la Figura 83, se evidencia la comunicación entre los dispositivos.

Figura 83

Prueba de la comunicación del robot y los dispositivos.



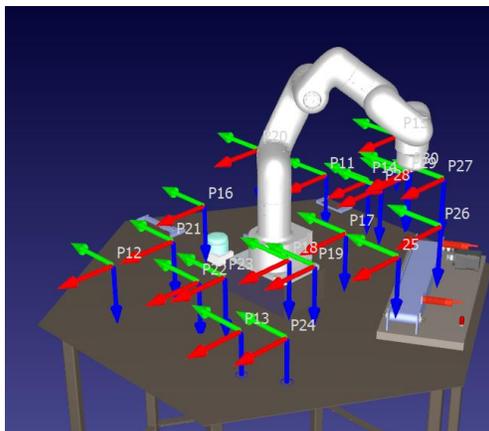
Prueba de la aplicación completa

Prueba de posición (XYZ)

En el layout diseñado en RoboDK se ingresaron los valores de varios puntos en diferentes posiciones (Figura 84), con la finalidad de validar el comportamiento del modelo virtual del robot en comparación del modelo real.

Figura 84

Posiciones configuradas en el layout de RoboDK.



En la Tabla 27 se detallan las coordenadas X, Y, Z de cada una de las posiciones que fueron ingresadas por el programador en la interfaz de RoboDK, en comparación con las coordenadas obtenidas en el controlador del robot UFACTORY Lite 6. Además, se presenta el cálculo del error para cada posición generada.

Tabla 27

Comparación de posiciones entre RoboDK y el controlador del robot.

N.	X		Error X	Y		Error Y	Z		Error Z
	Sim.	Ctr.		Sim.	Ctr.		Sim.	Ctr.	
P1	-345	-345,2	0,2	197	197,2	0,2	53,9	53,7	0,2
P2	-300	-300,3	0,3	150	150,4	0,4	200	200,2	0,2
P3	200	200,4	0,4	150	150,4	0,4	200	200	0
P4	120	120	0	250	250,2	0,2	50	49,6	0,4
P5	-55	-55	0	-300	-300,2	0,2	0	-0,4	0,4
P6	-300,6	-300,8	0,2	-156,8	-156,9	0,1	85,6	85,3	0,3
P7	-250,8	-251	0,2	316,5	316,7	0,2	135,5	135,3	0,2
P8	-90,8	-90,8	0	-13,5	-13,5	0	505,5	505,8	0,3
P9	9,2	9,2	0	-333,5	-333,8	0,3	85,5	85,2	0,3
P10	-300,8	-301	0,2	-243,5	-243,7	0,2	155,5	155,3	0,2
P11	-360,8	-361	0,2	-163,5	-163,6	0,1	65,5	65,3	0,2
P12	359,2	359,5	0,3	-163,5	-163,6	0,1	65,5	65,2	0,3
P13	309,2	309,3	0,1	206,5	206,6	0,1	25,5	25,1	0,4
P14	-380,8	-381,1	0,3	-33,5	-33,5	0	85,5	85,2	0,3
P15	-420,8	-421,1	0,3	-26,5	-26,5	0	215,5	215,4	0,1
P16	39,2	39,2	0	-203,5	-203,7	0,2	95,5	95,1	0,4
P17	-80,8	-80,9	0,1	156,5	156,7	0,2	105,5	105,1	0,4
P18	109,2	109,3	0,1	156,5	156,7	0,2	105,5	105,2	0,3
P19	89,2	89,3	0,1	216,5	216,7	0,2	105,5	105,1	0,4
P20	-190,8	-191	0,2	-243,5	-243,7	0,2	165,5	165,2	0,3
P21	219,2	219,4	0,2	-113,5	-113,6	0,1	95,5	95,2	0,3
P22	249,2	249,4	0,2	6,5	6,5	0	45,5	45,1	0,4
P23	279,2	279,5	0,3	126,5	126,6	0,1	115,5	115,2	0,3
P24	259,2	259,4	0,2	296,5	296,7	0,2	25,5	25,2	0,3

N.	X		Error X	Y		Error Y	Z		Error Z
	Sim.	Ctr.		Sim.	Ctr.		Sim.	Ctr.	
P25	-30,8	-30,8	0	356,5	356,8	0,3	135,5	135,3	0,2
P26	-150,8	-150,9	0,1	376,5	376,8	0,3	175,5	175,3	0,2
P27	-270,8	-271	0,2	286,6	286,7	0,1	225,5	225,4	0,1
P28	-250,8	-251	0,2	116,5	116,6	0,1	155,5	155,2	0,3
P29	-350,8	-351,1	0,3	116,5	116,6	0,1	155,5	155,3	0,2
P30	-230,8	-231	0,2	216,5	216,7	0,2	235,5	235,3	0,2
Error prom. X			0,17	Error prom. Y		0,17	Error prom. Z		0,27

Tras realizar 30 pruebas de posición se obtuvo como resultado que la magnitud del error absoluto existente que oscila entre 0 mm y 0.4 mm para cada eje. El error promedio fue de 0.17 mm, 0.17 mm y 0.27 mm para los ejes X, Y y Z respectivamente.

Para la aplicación de empaquetado el valor del error existente en los ejes, no es adecuado para generar los puntos de una trayectoria, por lo cual se necesitó compensar el valor de cada coordenada para cualquier posición en el layout desarrollado en RoboDK con el error promedio obtenido para cada eje, indicado en el párrafo anterior.

Prueba de posición (X, Y, Z) con las trayectorias realizadas

Con la compensación realizada a los ejes en la interfaz de RoboDK, se generaron 30 nuevas posiciones (Figura 85), para determinar el nuevo error existente en cada eje, los datos tabulados de estas pruebas se reflejan en la Tabla 28.

Figura 85

Posiciones compensadas.

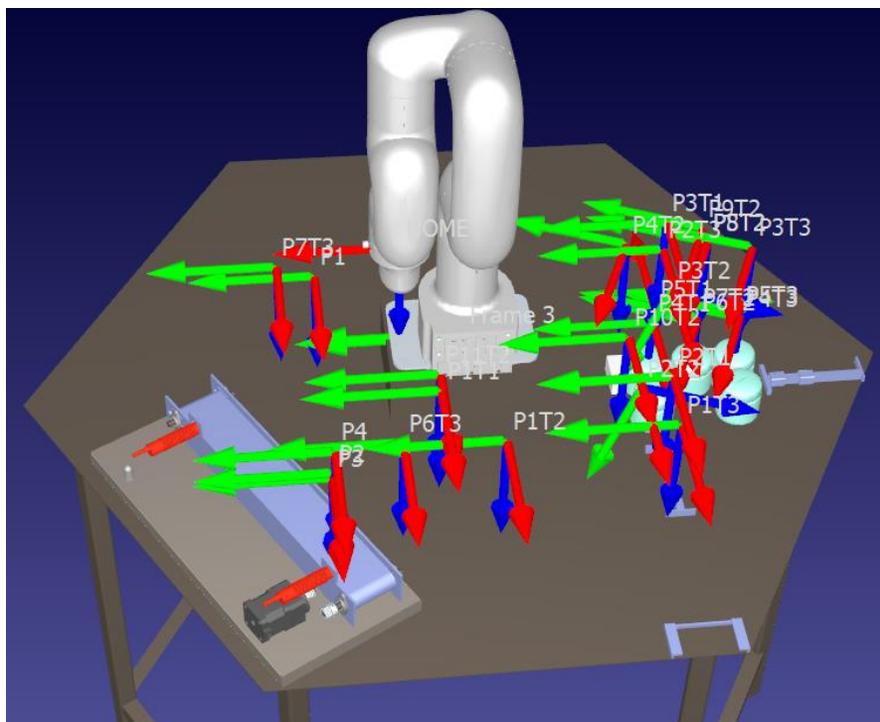


Tabla 28

Comparación de valores compensados y del controlador.

N.	X		Error	Y		Error	Z		Error	Error
	Sim.	Ctr.	X	Sim.	Ctr.	Y	Sim.	Ctr.	Z	XYZ
P1	0	0	0,0	195	194,9	0,1	115	115,4	0,4	0,17
P2	-345	-344,9	0,1	197	196,8	0,2	115	115,1	0,1	0,13
P3	-345	-344,3	0,3	198,00	198,2	0,2	107	107,3	0,3	0,27
P4	-355	-354,2	0,2	195	195,0	0,0	160	160,2	0,2	0,13
P5	-312	-311,7	0,3	71	70,8	0,2	203	203,2	0,2	0,23
P6	-312	-311,5	0,5	-197	-196,9	0,1	203	202,9	0,1	0,23
P7	-38	-38,0	0,0	-243	-242,9	0,1	203	203,2	0,2	0,10
P8	-38	-38,1	0,1	-243	-243,0	0,0	36	36,2	0,2	0,10
P9	-38	-38,0	0,0	-243	-242,8	0,2	60	60,2	0,2	0,13
P10	-330	-329,8	0,2	-197	-196,8	0,2	200	200,1	0,1	0,17
P11	-330	-329,8	0,2	71	70,7	0,3	200	199,8	0,2	0,23

N.	X		Y	Error Y		Z	Error Z		Error		
	Sim.	Ctr.		Sim.	Ctr.		Sim.	Ctr.	XYZ		
P12	-20,8	-20,8	0,0	-199,6	-199,4	0,2	157,3	157,6	0,3	0,17	
P13	-28	-28,0	0,0	-303	-303,0	0,0	160	160,2	0,2	0,07	
P14	-42,7	-42,6	0,1	-300,6	-300,5	0,1	40	39,7	0,3	0,17	
P15	-42,7	-42,6	0,1	-300,6	-300,6	0,0	45	45,2	0,2	0,10	
P16	-28	-28,0	0,0	-303	-302,9	0,1	160	160,2	0,2	0,10	
P17	-163,4	-163,2	0,2	199,6	199,4	0,2	154,4	154,6	0,2	0,20	
P18	-348	-347,8	0,2	-204,4	-204,2	0,2	160	160,2	0,2	0,20	
P19	-38	-38,0	0,0	-243	-242,8	0,2	160	160,3	0,3	0,17	
P20	-42,7	-42,9	0,2	-358	-357,7	0,3	160	160,2	0,2	0,23	
P21	-42,7	-42,9	0,2	-358	-357,8	0,2	40	39,8	0,2	0,20	
P22	-42,7	-42,9	0,2	-358	-357,8	0,2	45	44,7	0,3	0,23	
P23	-348	-347,7	0,3	117	117,0	0,0	160	160,1	0,1	0,13	
P24	-31,2	-31,1	0,1	242,7	242,7	0,0	160	160,3	0,3	0,13	
P25	-101,9	-101,9	0,0	-240,5	-240,3	0,2	160	160,2	0,2	0,13	
P26	-102	-102,0	0,0	-240,4	-240,2	0,2	39,8	40,1	0,3	0,17	
P27	-102,1	-102,1	0,0	-240,6	-240,4	0,2	45	45,4	0,4	0,20	
P28	-101	-101,1	0,1	-262,2	-262,1	0,1	160	159,8	0,2	0,13	
P29	-98	-97,9	0,1	-358,1	-357,8	0,3	160	159,7	0,3	0,23	
P30	-98	-97,8	0,2	-358,1	-357,9	0,2	40	40,0	0,0	0,13	
Error promedio			0,13				0,15			0,24	0,16

Nota. La etiqueta Sim. significa Simulado y la etiqueta Ctr. significa Controlador

Con los resultados obtenidos de la Tabla 28, se obtuvo un error promedio de 0.15 mm, 0.15 mm y 0.24 mm respectivamente para los ejes X, Y y Z. De estos resultados se obtiene valores del error XYZ los cuales se categorizan en: satisfactorio con un error menor a 0.15, bueno con un error de hasta 0.19 mm y regular con un error superior o igual 0.2 mm.

Pruebas de trayectorias

En el entorno de simulación de RoboDK se tiene la posibilidad de realizar 3 tipos de movimientos: movimiento articular (Figura 86), movimiento lineal (Figura 88) y movimiento

circular (Figura 90), mismos que fueron validados por el parámetro de posición, velocidad o forma según el caso.

- *Trayectoria articular:* En la Figura 86, se realizó la simulación de la trayectoria y en la Figura 87 se ejecutó el trayecto en el entorno físico de la celda, la verificación de esta trayectoria se lo realizó por el parámetro de forma.

Figura 86

Movimiento articular en RoboDK.

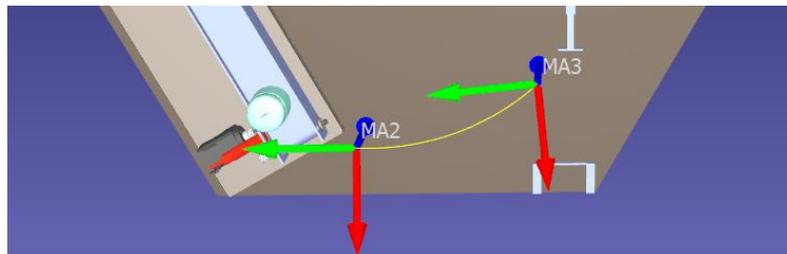
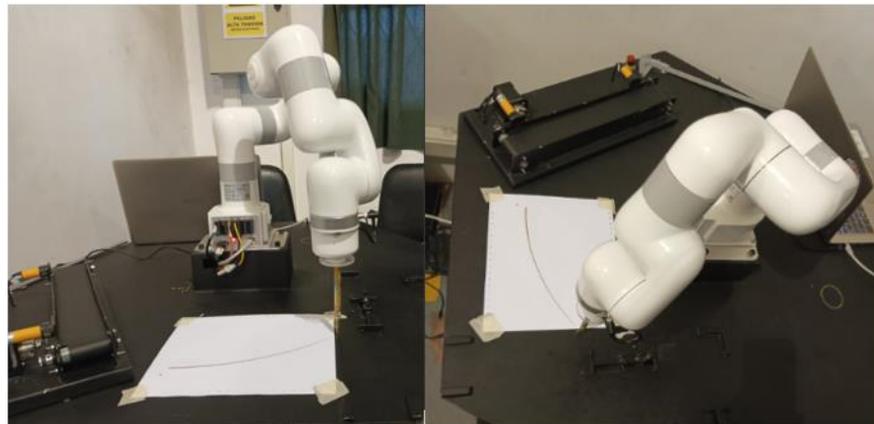


Figura 87

Movimiento articular ejecutado en la celda.



- *Trayectoria lineal:* En la Figura 88 se realiza la simulación de la trayectoria y en la Figura 89 se muestra la ejecución en el entorno real de trabajo. Los resultados se clasifican en:

satisfactorio con un error de 0, bueno con un error de hasta 0.1mm y regular con un error superior a 0.1mm.

Figura 88

Movimiento lineal en RoboDK.

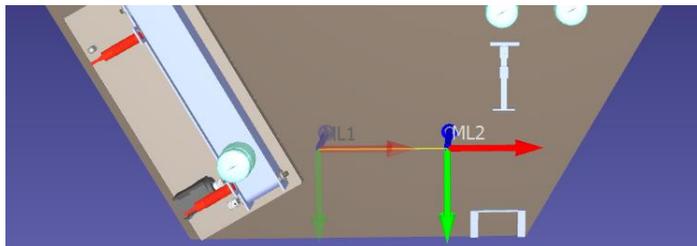
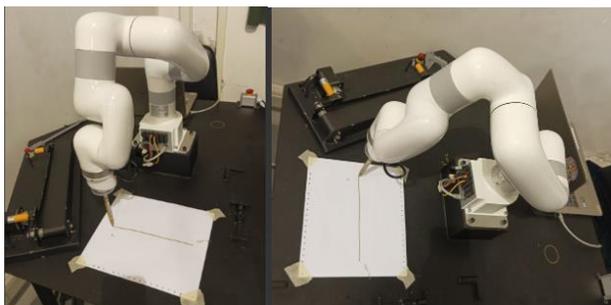


Figura 89

Movimiento lineal ejecutado en la celda.



La Tabla 29 evidencia los datos de la distancia del simulador, en comparación con datos de distancia real medidos con un calibrador; comprobando que las distancias en el simulador y en el entorno real son equivalentes. Los valores obtenidos fueron evaluados en el eje Y.

Tabla 29

Datos de desplazamiento en Y.

N	Desplazamiento en Y		
	D. Sim.	D. Real	Error
1	40	40,2	0,2
2	45	45	0
3	50	50,2	0,2

N	Desplazamiento en Y		
	D. Sim.	D. Real	Error
4	55	55,1	0,1
5	60	60,2	0,1
6	65	65	0
7	70	70	0
8	75	75	0
9	80	80,1	0,1
10	85	85,1	0,1
11	90	90,2	0,2
12	95	95	0
13	100	100,2	0,2
14	105	105	0
15	110	110,1	0,1
16	115	115,1	0,1
17	120	120,1	0,1
18	125	125,1	0,1
19	130	130,1	0,1
20	135	135,2	0,2
21	140	140	0
22	145	145	0
23	150	150,2	0,2
24	155	155,1	0,1
25	160	160,1	0,1
26	170	170,1	0,1
27	180	180,2	0,2
28	190	190,2	0,2
29	200	200,1	0,1
30	210	210,2	0,2

Nota. La etiqueta D. Sim. significa Diámetro Simulado y la etiqueta D. Real significa Diámetro Real.

La Tabla 30 muestra la comparación de los puntos inicial y final en las coordenadas X y Z obtenidos en el entorno simulado y real, debido a que, los puntos de dichas coordenadas son invariables, se comprobó la ejecución de una trayectoria lineal.

Tabla 30

Comparación entre punto inicial y final de las coordenadas X y Z.

N	Coordenada X		Coordenada Z	
	PI – PF (Sim.)	PI – PF (Ctrl.)	PI – PF (Sim.)	PI – PF (Ctrl.)
1	-355	-355,3	38,2	38,3
2	-355	-355,1	38,2	38,3
3	-355	-355	38,2	38,3
4	-355	-355,1	38,2	38,2
5	-355	-355,3	38,2	38,5
6	-355	-355,3	38,2	38,2
7	-355	-355,2	38,2	38,5
8	-355	-355,2	38,2	38,3
9	-355	-355,3	38,2	38,3
10	-355	-355,3	38,2	38,5
11	-355	-355,1	38,2	38,5
12	-355	-355	38,2	38,3
13	-355	-355,3	38,2	38,4
14	-355	-355,1	38,2	38,4
15	-355	-355,2	38,2	38,5

- *Trayectoria circular:* En la Figura 90 se realizó la simulación de la trayectoria y en la Figura 91 se muestra la ejecución de dicha trayectoria en el cobot.

Figura 90

Movimiento Circular

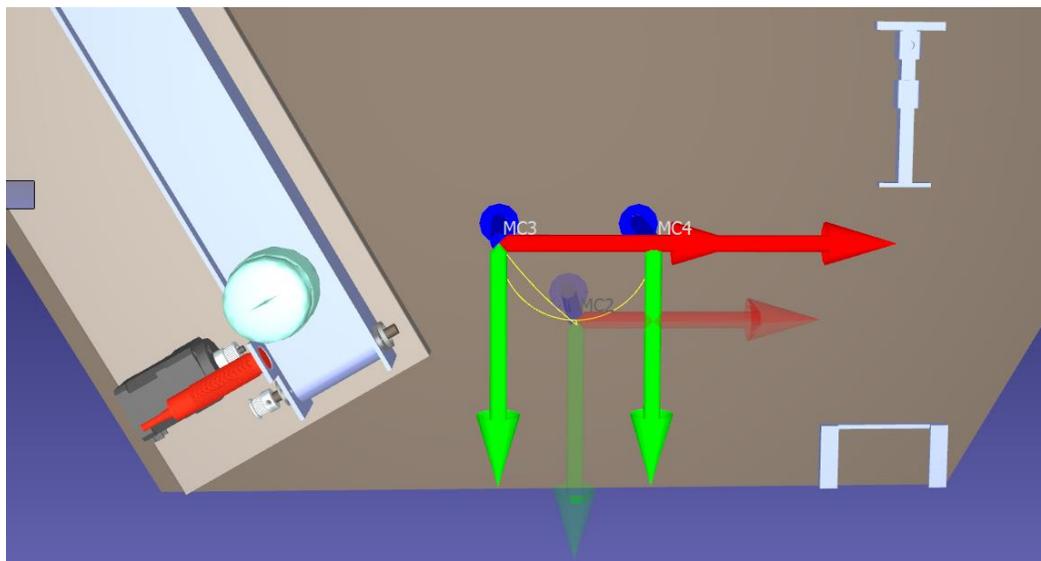


Figura 91

Movimiento circular ejecutado en la celda.



La Tabla 31 evidencia los valores de los radios y diámetros de los ensayos simulados y reales de semicircunferencias creadas para la comprobación de una trayectoria circular. Cabe

mencionar que de manera física se realizó la medición de los radios y diámetros con un calibrador. En los ensayos realizados se consideró siempre el punto inicial arbitrario (-305, -50, 42) y se modificó el punto final arbitrario para la obtención de diferentes diámetros, considerando dicha variación en el eje X hasta un máximo de -355.

Tabla 31

Valores de radio y diámetros simulados y reales.

N	Radio		Error	Diámetro		Error	Error
	R. Sim.	R. Medido	R	D. Sim.	D. Medido	D	General
1	10	10	0	20	20,2	0,2	0,1
2	11	11,2	0,2	22	22,2	0,2	0,2
3	12	12,1	0,1	24	24	0	0,05
4	14	14,1	0,1	28	28,1	0,1	0,1
5	15	15,1	0,1	30	30,2	0,2	0,15
6	16	16,1	0,1	32	32,1	0,1	0,1
7	17	17	0	34	34	0	0
8	19	19,2	0,2	38	38,2	0,2	0,2
9	20	20	0	40	40,1	0,1	0,05
10	22	22	0	44	44,2	0,2	0,1
11	23	23	0	46	46,1	0,1	0,05
12	24	24,1	0,1	48	48,1	0,1	0,1
13	25	25,2	0,2	50	50,1	0,1	0,15
14	27	27,2	0,2	54	54,2	0,2	0,2
15	28	28	0	56	56	0	0
16	29	29	0	58	58,2	0,2	0,1
17	30	30	0	60	60,1	0,1	0,05
18	32	32,1	0,1	64	64	0	0,05
19	33	33	0	66	66,1	0,1	0,05
20	34	34,2	0,2	68	68,2	0,2	0,2
21	35	35,2	0,2	70	70,2	0,2	0,2
22	37	37,1	0,1	74	74,2	0,2	0,15
23	39	39,2	0,2	78	78,1	0,1	0,15

N	Radio		Error	Diámetro		Error	Error
	R. Sim.	R. Medido	R	D. Sim.	D. Medido	D	General
24	40	40	0	80	80,2	0,2	0,1
25	42	42,2	0,2	84	84,1	0,1	0,15
26	44	44	0	88	88,2	0,2	0,1
27	45	45,1	0,1	90	90,1	0,1	0,1
28	47	47,1	0,1	94	94	0	0,05
29	49	49,2	0,2	98	98,1	0,1	0,15
30	50	50,2	0,2	100	100	0	0,1

Nota. La etiqueta R. Sim. significa Radio Simulado, la etiqueta R. Medido significa Radio Medido, la etiqueta D. Sim. significa Diámetro Simulado y la etiqueta D. Medido significa Diámetro Medido,

Si se revisan los datos obtenidos en la Tabla 31 se evidencia que la variación del radio o del diámetro es mínimo en comparación al simulado en RoboDK, por lo que, se comprobó la ejecución de una correcta trayectoria circular. El error general es el promedio entre el error R y el error D, los resultados de este promedio se clasifican en: satisfactorio con un error menor de 0.05, bueno con un error de hasta 0.15 mm y regular con un error de a 0.2 mm.

Pruebas de velocidad y repetibilidad

Tras la verificación de las trayectorias y comprobación de la ausencia de colisiones entre los accesorios y el robot, se procede a realizar pruebas de velocidad y repetibilidad, en una trayectoria lineal programada el software RoboDK, las posiciones de punto inicial (Figura 92) y punto final (Figura 93) describen una distancia de 500 mm y una velocidad programada de 500 mm/s, como se muestra en la Figura 94.

Figura 92

Punto inicial de la trayectoria sometida a pruebas de velocidad y repetibilidad.

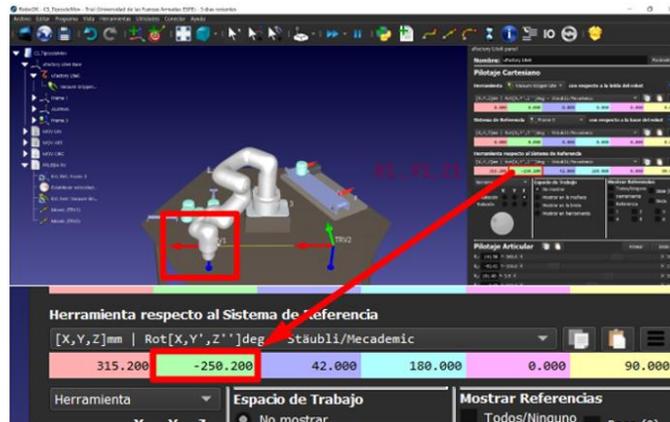


Figura 93

Punto final de la trayectoria sometida a pruebas de velocidad y repetibilidad.

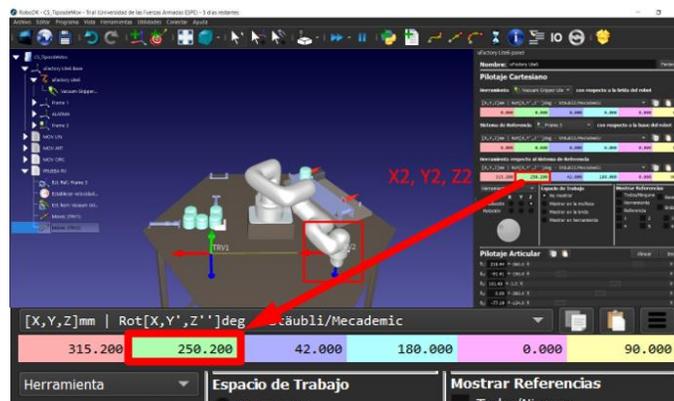
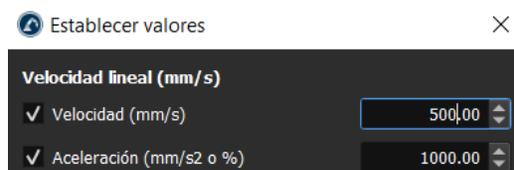


Figura 94

Parámetros de velocidad configurada en RoboDK.



La Tabla 32 contiene los datos de las pruebas de velocidad que se tomaron de manera manual, los mismos que fueron medidos entre dos puntos a una distancia constante, utilizando un cronómetro. Donde aplicando la ecuación básica de velocidad (distancia/tiempo) se obtuvo

una velocidad promedio de 498.97 mm/s, debido a esto se verificó que la velocidad que se configura el software RoboDK es similar a la que trabaja el robot en el entorno real.

Tabla 32

Datos para pruebas de velocidad.

Datos obtenidos				
N	Tiempo	Distancia	Velocidad	Error V.
1	1,02	500	490,20	9,8
2	0,97	500	515,46	15,46
3	1,01	500	495,05	4,95
4	1,01	500	495,05	4,95
5	0,99	500	505,05	5,05
6	1,02	500	490,20	9,8
7	1	500	500,00	0
8	0,97	500	515,46	15,46
9	0,98	500	510,20	10,2
10	0,99	500	505,05	5,05
11	1	500	500,00	0
12	0,97	500	515,46	15,46
13	0,98	500	510,20	10,2
14	1,01	500	495,05	4,95
15	1,01	500	495,05	4,95
16	0,97	500	515,46	15,46
17	0,99	500	505,05	5,05
18	1,01	500	495,05	4,95
19	0,99	500	505,05	5,05
20	0,98	500	510,20	10,2
21	0,99	500	505,05	5,05
22	0,97	500	515,46	15,46
23	1,02	500	490,20	9,8
24	0,99	500	505,05	5,05
25	1,01	500	495,05	4,95
26	1,01	500	495,05	4,95

Datos obtenidos				
N	Tiempo	Distancia	Velocidad	Error V.
27	0,98	500	510,20	10,2
28	1,02	500	490,20	9,8
29	0,97	500	515,46	15,46
30	0,98	500	510,20	10,2
Velocidad Promedio			498,97	

En la Tabla 32 el Error V. es el error absoluto entre la velocidad medida y la velocidad real, los resultados del error se clasifican en: satisfactorio con un error menor de 5.05, bueno con un error de hasta 10.2 mm y regular con un error de a 15.46 mm.

Adicionalmente, en la Tabla 33 se muestran los ensayos de repetibilidad en la trayectoria antes mencionada en un bucle de 30 repeticiones y medidas con un calibrador a lo largo del eje Y (las coordenadas del eje X y eje Z son constantes), con lo que se evidenció que el brazo robótico siempre cumple con las posiciones del trayecto, obteniendo un error promedio de 0.06 mm, además los resultados del error se clasifican en: satisfactorio hasta 0.05, bueno hasta 0.08 y regular mayor 0.08.

Tabla 33

Datos para pruebas de velocidad.

N	Repetibilidad		
	Distancia	Lectura	Error
1	500	500,06	0,06
2	500	500,09	0,09
3	500	500,07	0,07
4	500	500,03	0,03
5	500	500,04	0,04
6	500	500,07	0,07
7	500	500,10	0,10
8	500	500,09	0,09

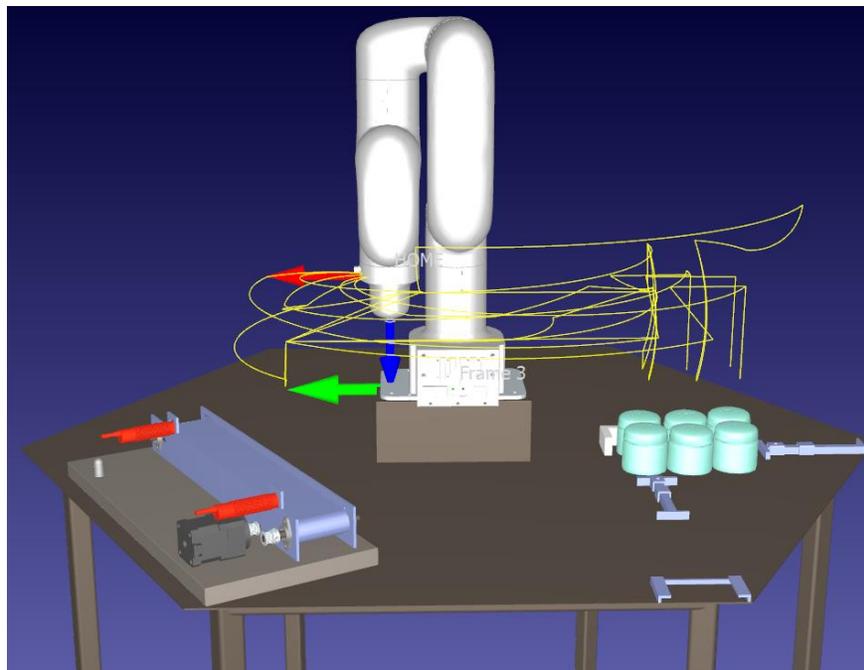
N	Repetibilidad		
	Distancia	Lectura	Distancia
9	500	500,02	0,02
10	500	500,05	0,05
11	500	500,02	0,02
12	500	500,03	0,03
13	500	500,08	0,08
14	500	500,00	0,00
15	500	500,02	0,02
16	500	500,07	0,07
17	500	500,01	0,01
18	500	500,07	0,07
19	500	500,07	0,07
20	500	500,07	0,07
21	500	500,03	0,02
22	500	500,10	0,10
23	500	500,10	0,10
24	500	500,07	0,07
25	500	500,08	0,08
26	500	500,07	0,06
27	500	500,06	0,06
28	500	500,05	0,05
29	500	500,01	0,01
30	500	500,09	0,09
Prom. de la repetibilidad			0,06

Una vez comprobados los tipos de trayectorias generadas en RoboDK, realizadas las pruebas de velocidad, repetibilidad y ejecutadas en el ambiente de trabajo real, se procede con la programación fuera de línea de las trayectorias necesarias; puesto que, los resultados de las pruebas parciales son satisfactorios se desarrollan las trayectorias para la aplicación de empaquetado de productos (Figura 95), que posteriormente son cargadas en el controlador del

cobot; llegando a la conclusión de que los valores de la aplicación simulada son equivalentes a los valores de la ejecución en el entorno real.

Figura 95

Trayectorias para la aplicación de empaquetado.



Validación de la hipótesis

Validación didáctica de la celda robotizada

La celda robotizada fue sometida a pruebas con estudiantes de noveno nivel paralelo A de la carrera de Mecatrónica de la Universidad de las Fuerzas Armadas ESPE Sede Latacunga, donde los estudiantes recibieron una inducción sobre la manipulación de la celda y la metodología de programación fuera de línea, para la creación de trayectorias en la aplicación de empaquetado de productos; al culminar se realizaron 22 encuestas con la finalidad de emplear el método de investigación descriptivo que permite estudiar el resultado de cada pregunta aplicada.

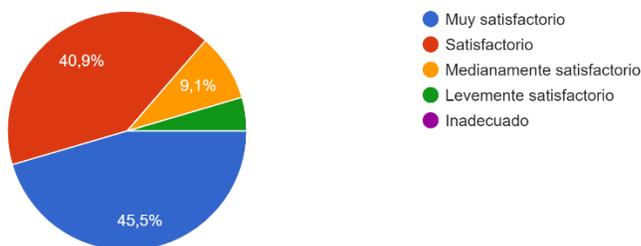
Las respuestas obtenidas en las encuestas, son representadas en gráficas de sectores, la encuesta aplicada se encuentra en el Anexo 3.

Con el fin de analizar que el uso de la celda robotizada fomenta el espíritu de investigación y diseño de celdas, se presenta los resultados de la Figura 96, donde el 86.4% del alumnado encuestado responden entre muy satisfactorio y satisfactorio, el 9.1% medianamente satisfactorio y el 4.5% levemente satisfactorio; obteniendo un 100% de aceptación al no obtener una respuesta negativa.

Figura 96

Resultados de la pregunta 1.

1. ¿Cree usted qué con el uso del brazo robótico y accesorios de la celda robotizada para trabajar la robótica industrial en el ámbito educativo, permiti...a investigación y el diseño de celdas robotizadas?
22 respuestas



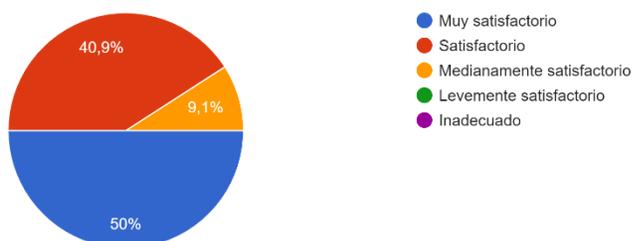
La Figura 97, muestra que el 50% de los participantes encuestados responden con muy satisfactorio; el 40.9% satisfactorio y el 9.1% medianamente satisfactorio; logrando el 100% de aceptación al no obtener una respuesta negativa, definiendo a la celda robotizada como una herramienta para la enseñanza de la programación fuera de línea.

Figura 97

Resultados de la pregunta 2.

2. ¿La celda robotizada puede considerarse como una herramienta para el proceso de enseñanza en el método de programación offline?

22 respuestas



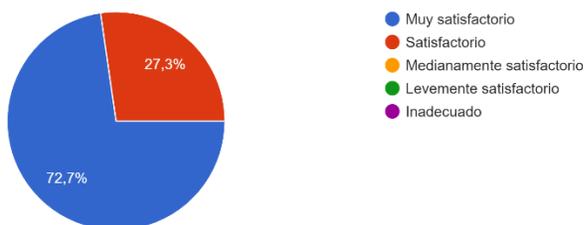
En el diagrama de la Figura 98, se obtiene el 100% de respuestas positivas donde los participantes encuestados responden entre muy satisfactorio y satisfactorio; determinando la aceptación para crear guías de aprendizaje sobre la metodología de programación fuera de línea.

Figura 98

Resultados de la pregunta 3.

3. ¿Considera que es de utilidad elaborar una guía sobre la enseñanza de la programación offline en la celda robotizada como un recurso de aprendizaje?

22 respuestas



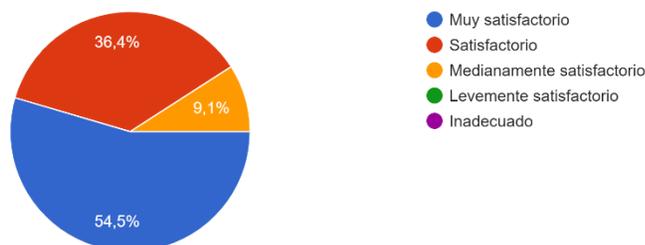
Al analizar los resultados de la Figura 99, se consigue el 90.9% de respuestas entre muy satisfactorio y satisfactorio; el 10.5% responde con medianamente satisfactorio, ante la facilidad de comprender el desarrollo de las actividades relacionadas con robótica industrial.

Figura 99

Resultados de la pregunta 4.

4. ¿Con la manipulación de la celda robotizada al alumno le resulta fácil entender el desarrollo de las actividades relacionados con robótica industrial?

22 respuestas



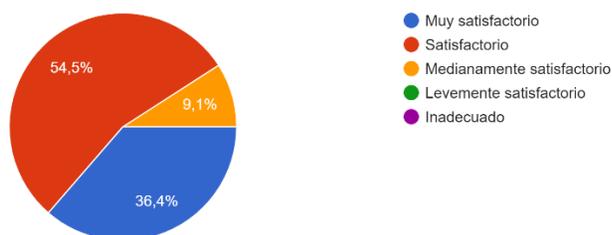
La Figura 100 indica que la programación fuera de línea aplicada en la celda robotizada permite cumplir con los objetivos curriculares en la etapa de educación, tiene un 100% de aceptación entre los estudiantes con respuestas en los niveles de muy satisfactorio, satisfactorio y medianamente satisfactorio.

Figura 100

Resultados de la pregunta 5.

5. ¿La celda robotizada con programación offline permite el alcance de los objetivos curriculares de la etapa de educación?

22 respuestas



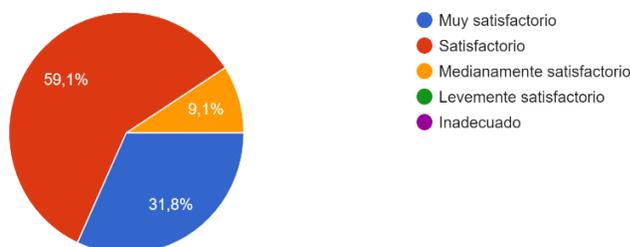
Los resultados de la Figura 101 arrojan que el 59.1% de los estudiantes encuestados seleccionan la categoría muy satisfactoria al considerar a la celda robotizada como una herramienta en el desarrollo del pensamiento lógico y matemático, el restante 40.9% de los

encuestados aceptan la premisa con respuestas entre muy satisfactorio y medianamente satisfactorio; al no percibir respuestas negativas se tiene un 100% de aceptación.

Figura 101

Resultados de la pregunta 6.

6. ¿La celda robotizada es una herramienta útil para el desarrollo del pensamiento lógico y matemático en los alumnos?
22 respuestas

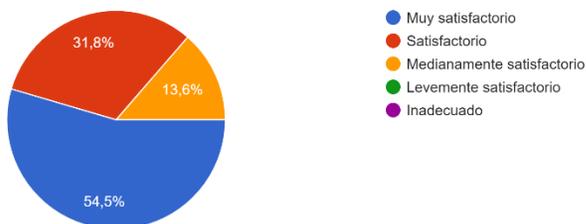


En la Figura 102 se tiene el 86.3% de contestación entre los niveles de muy satisfactorio a satisfactorio, el 13.6% contesta con el nivel medianamente satisfactorio; ante la cuestión del ambiente de cooperatividad entre el operario y el brazo robótico; con el análisis aplicado se alcanza el 100% de aceptación por parte del alumnado.

Figura 102

Resultados de la pregunta 7.

7. ¿La celda robotizada permite evidenciar el trabajo cooperativo entre el operario y el brazo robótico?
22 respuestas



El nivel de satisfacción expresado por los estudiantes de noveno nivel ante la implementación de la celda robotizada para programación fuera de línea es del 59.1% en el

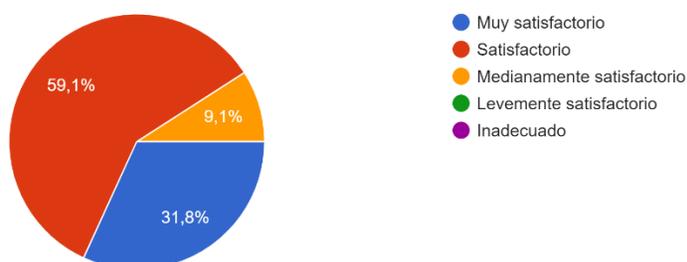
nivel de muy satisfactorio, el 31.8% manifiesta que fue satisfactorio y el 9.1% dice que es medianamente satisfactorio; obteniendo un 100% de aceptación en la implementación de la celda (Figura 103).

Figura 103

Resultados de la pregunta 8.

8. ¿Existe un alto nivel de satisfacción en cuanto a los resultados obtenidos con la implementación de la celda robotizada para programación offline?

22 respuestas



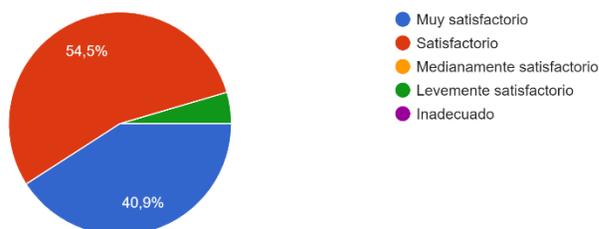
La Figura 104 da como resultado que el 91.4% de los estudiantes consideran que implementar la celda robotizada para programación fuera de línea influye positivamente al rendimiento académico y el 4.5% manifiesta que es levemente satisfactorio la implementación del proyecto; obteniendo una gran acogida del proyecto.

Figura 104

Resultados de la pregunta 9.

9. ¿Considera que la implementación de la celda robotizada para programación offline influye positivamente en la mejora del rendimiento académico del alumnado?

22 respuestas

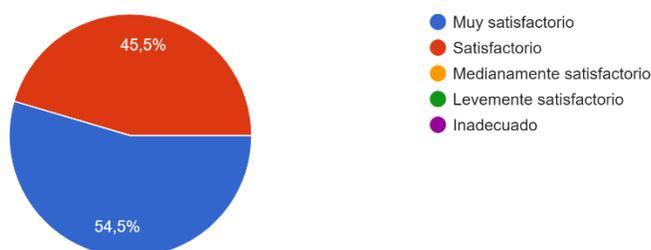


Los resultados de la Figura 105 dan a conocer que el 100% del alumnado concuerda en que la programación fuera de línea aplicada en la celda robotizada es una herramienta atractiva para el programa de enseñanza, respondiendo entre los niveles de muy satisfactorio con un 45.5% y satisfactorio de 54.5%.

Figura 105

Resultados de la pregunta 10.

10. ¿La celda robotizada para programación offline es una herramienta atractiva que motiva al alumnado en el aprendizaje?
22 respuestas



Al culminar el análisis de los datos obtenidos mediante la aplicación de encuestas, se logra concluir el cumplimiento de los parámetros planteados en el desarrollo del proyecto, logrando de esta manera que los estudiantes a través de la manipulación de la celda robotizada eleven su interés por la investigación de nuevas metodologías de programación para robótica industrial y obteniendo un crecimiento en los niveles de aprendizaje.

Validación del control de trayectorias en el brazo robótico antropomórfico

Para validar la hipótesis se usa el método estadístico Chi-cuadrado, tomando en cuenta que la hipótesis planteada es: ¿El diseño e implementación de una celda robotizada didáctica integrada con software de programación off-line permitirá controlar las trayectorias del brazo robótico antropomórfico, en una aplicación de empaquetado de productos?

Planteamiento de Hipótesis

- (Ho)= El diseño e implementación de una celda robotizada didáctica integrada con software de programación off-line no permitirá controlar las trayectorias del brazo robótico antropomórfico, en una aplicación de empaquetado de productos.
- (Hi)= El diseño e implementación de una celda robotizada didáctica integrada con software de programación off-line permitirá controlar las trayectorias del brazo robótico antropomórfico, en una aplicación de empaquetado de productos.

La Tabla 34 alberga los ensayos y resultados obtenidos en las pruebas realizadas en el Capítulo V Pruebas y Resultados, para la evaluación de la hipótesis; los resultados son categorizados en tres niveles: satisfactorio para un error de 0, bueno para un error de hasta 0.1mm y regular para un error superior a 0.1mm.

Tabla 34

Pruebas realizadas.

Ítem	Descripción	Satisfactorio	Bueno	Regular	# Pruebas
1	Posicionamiento XYZ	13	6	11	30
2	Trayectoria Lineal	8	10	12	30
3	Trayectoria circular	9	10	11	30
4	Velocidad	15	9	6	30
5	Repetibilidad	12	10	8	30
Total		57	45	48	150

Se realiza el cálculo de la frecuencia esperada necesaria (Ecuación 3) para aplicar el método de chi-cuadrado.

$$f_t = \frac{T_i * T_j}{N} \quad \text{Ecuación 3}$$

Donde:

- T_i = Número de pruebas
- T_j = Total de satisfactorias, bueno o regular.
- N = Total de pruebas

Entonces al calcular la frecuencia esperada satisfactorio, se tiene:

$$\text{Frecuencia esperada satisfactorio} = \frac{30 * 57}{150} = 11,4$$

La frecuencia esperada bueno es:

$$\text{Frecuencia esperada bueno} = \frac{30 * 56}{150} = 11,2$$

La frecuencia esperada regular es:

$$\text{Frecuencia esperada regular} = \frac{30 * 37}{150} = 5,55$$

Para calcular el valor de chi-cuadrado se aplica la fórmula de la Ecuación 4:

$$X^2 = \sum \frac{(f_o * f_t)^2}{f_t} \quad \text{Ecuación 4}$$

Donde:

- X^2 = valor chi - cuadrado
- f_o = frecuencia absoluta
- f_t = frecuencia esperada

La Tabla 35 muestra cada uno de los valores necesarios para el cálculo de chi cuadrado y el valor obtenido a comparar.

Tabla 35

Cálculo del método estadístico.

Datos	f_o	f_t	$(f_o - f_t)$	$(f_o - f_t)^2$	$\frac{(f_o - f_t)^2}{f_t}$
Satisfactorio 1	13	11,4	1,6	2,56	0,2246
Satisfactorio 2	8	11,4	-3,4	11,56	1,0140
Satisfactorio 3	9	11,4	-2,4	5,76	0,5053
Satisfactorio 4	15	11,4	3,6	12,96	1,1368
Satisfactorio 5	12	11,4	0,6	0,36	0,0316
Bueno 1	6	11,2	-5,2	27,04	2,4143
Bueno 2	13	11,2	1,8	3,24	0,2893
Bueno 3	16	11,2	4,8	23,04	2,0571
Bueno 4	9	11,2	-2,2	4,84	0,4321
Bueno 5	12	11,2	0,8	0,64	0,0571
Regular 1	11	5,55	5,45	29,7025	5,3518
Regular 2	9	5,55	3,45	11,9025	2,1446
Regular 3	5	5,55	-0,55	0,3025	0,0545
Regular 4	6	5,55	0,45	0,2025	0,0365
Regular 5	6	5,55	0,45	0,2025	0,0365
				Total	16,7497

Se calcula los grados de libertad con la Ecuación 5:

$$(r - 1) * (c - 1)$$

Ecuación 5

Donde:

r = # de filas

c = # de columnas

$$(5 - 1) * (3 - 1) = 8$$

En la Figura 106 se busca el valor de "p" en la categoría 0,05% de error permitido.

Figura 106

Selección del valor de nivel de confianza en función del error.

Grados de libertad	Áreas en la cola superior									
	0.995	0.99	0.975	0.95	0.90	0.10	0.05	0.025	0.01	0.005
1	0.000	0.000	0.001	0.004	0.016	2.706	3.841	5.024	6.635	7.879
2	0.010	0.020	0.051	0.103	0.211	4.605	5.991	7.378	9.210	10.597
3	0.072	0.115	0.216	0.352	0.584	6.251	7.815	9.348	11.345	12.838
4	0.207	0.297	0.484	0.711	1.064	7.779	9.488	11.143	13.277	14.860
5	0.412	0.554	0.831	1.145	1.610	9.236	11.070	12.832	15.086	16.750
6	0.676	0.872	1.237	1.635	2.204	10.645	12.592	14.449	16.812	18.548
7	0.989	1.239	1.690	2.167	2.833	12.017	14.067	16.013	18.475	20.278
8	1.344	1.647	2.180	2.733	3.490	13.362	15.507	17.535	20.090	21.955
9	1.735	2.088	2.700	3.325	4.168	14.684	16.919	19.023	21.666	23.589

Nota. Se selecciona el valor crítico en función del nivel de confianza y grados de libertad.

Tomado de (Escobar, 2022)

Para un valor de confianza de 95% se tiene un valor crítico de 15.507, se debe tomar en cuenta que el valor calculado es de 16,7497 al ser mayor da como resultado el rechazo de la hipótesis nula y la aprobación de la hipótesis alternativa con 95% de confiabilidad, la cual menciona que: *El diseño e implementación de una celda robotizada didáctica integrada con software de programación off-line permitirá controlar las trayectorias del brazo robótico antropomórfico, en una aplicación de empaquetado de productos.*

Capítulo VI

Conclusiones y Recomendaciones

Conclusiones

Con el diseño e implementación de una celda robotizada didáctica en el laboratorio de Robótica Industrial de la Universidad de las Fuerzas Armadas ESPE Sede Latacunga se concluye que mediante la utilización del software RoboDK con la metodología de programación fuera de línea se logró el adecuado control de trayectorias en el cobot UFACTORY Lite 6 enfocado en la aplicación de empaquetado de productos.

Mediante una investigación bibliográfica sobre robots antropomórficos y sus aplicaciones actuales se obtiene una visión actualizada de las tendencias y avances en este campo. Esta indagación lleva a la elección de un nuevo modelo de robot para la celda robotizada: el cobot modelo Lite 6 de la empresa UFACTORY.

El robot seleccionado posee 6 grados de libertad, es de tipo articular, con una capacidad de carga de 1kg, un alcance de 440 mm, es compacto, permite diseñar aplicaciones en el lenguaje de programación Python, además, cuenta con dos métodos de programación por guiado y textual; disponibles en la interfaz UFACTORY Studio, de uso intuitivo.

El cobot UFACTORY Lite 6 dispone de tres tipos de movimientos: lineal, articular y circular, los cuales pueden ser ejecutados a una velocidad máxima de 500 mm/s en el TCP y una repetibilidad de ± 0.2 mm según datos del fabricante.

En el proceso de contrastar y evaluar los programas para la programación fuera de línea compatibles con el brazo robótico seleccionado, se adquiere el software RoboDK con una licencia estudiantil, considerando la disponibilidad de una amplia gama de librerías para robots industriales, entre estos KUKA, ABB, FANUC y otros; además de librerías para robots

colaborativos como es el caso del cobot del presente proyecto, lo que asegura la compatibilidad técnica y operativa del robot.

Para el desarrollo de la programación fuera de línea en RoboDK se obtiene un error promedio absoluto entre los valores simulados y reales de: 0.13 mm, 0.15 mm y 0.24 mm en los ejes X, Y y Z, respectivamente; además, se valida la trayectoria angular, lineal y circular generada en el software; verificando que la velocidad programada en el entorno virtual es equivalente a la del entorno real, lo que da como resultado la validación del control de trayectorias generadas en RoboDK para la aplicación de empaquetado.

El software RoboDK dispone de una gran variedad de Post-procesadores para robots industriales y cobots, el cual genera el código de programación para una aplicación específica y su posterior implementación en el controlador del mismo; tal es el caso de la aplicación de empaquetado.

Para el diseño y construcción de la estructura mecánica de la celda robotizada, se considera la norma NTE INEN 1 641 para la determinación de sus dimensiones y requerimientos. La disposición del área de trabajo es definida por un layout centralizado, materializando un entorno funcional para la aplicación de empaquetado, incluyendo parámetros de seguridad visuales y eléctrico, enfocándose en salvaguardar la integridad del operario en el ambiente de trabajo compartido con el cobot.

El diseño de la estructura de la banda transportadora asegura un transporte adecuado a una velocidad constante de 50 mm/s del producto, para la aplicación de empaquetado; para su dimensionamiento se considera el espacio de trabajo disponible dentro de la celda; además, la implementación de un control ON-OFF que permite una sincronización adecuada con el ritmo de producción, garantizando un proceso fluido y efectivo.

El ensamblaje exitoso del cobot UFACTORY Lite 6 con la estructura mecánica diseñada, da como resultado la materialización de la celda robotizada propuesta, lo que permite obtener los parámetros necesarios para crear el entorno de trabajo virtual en el software RoboDK; consecuentemente se logra configurar el sistema de comunicaciones entre los accesorios de la celda y el robot bajo un sistema de control descentralizado.

Con la utilización del método estadístico del chi cuadrado aplicado a los resultados de las pruebas realizadas se obtiene como conclusión la aceptación de la hipótesis alternativa y rechazo de la hipótesis nula. Mediante encuestas aplicadas al alumnado de Robótica Industrial se concluye que la celda robotizada es didáctica con un 83.62% de aceptación.

Recomendaciones

Para aplicaciones educativas, de investigación y colaborativas se recomienda el uso del cobot UFACTORY Lite 6; debido a: facilidad de programación, seguridad en el entorno de trabajo colaborativo con el operario, operación intuitiva de la interfaz del controlador, compatibilidad con el software RoboDK para programación fuera de línea.

Antes de manipular la celda robotizada y sus componentes se recomienda realizar previamente lectura del manual de usuario del robot UFACTORY Lite 6 del fabricante, para familiarizarse con el funcionamiento del cobot.

Se sugiere el uso del software RoboDK, debido a que dispone una amplia gama de librerías para robots industriales, entre estos KUKA, ABB, FANUC y para robots colaborativos como: UFACTORY, UR, entre otros; además que cuenta con un entorno intuitivo para realizar la programación fuera de línea de cualquier aplicación. Además, se aconseja realizar una lectura de la documentación técnica del software RoboDK, para un mayor conocimiento de las funcionalidades.

Previo a la carga del archivo en el controlador del robot, se recomienda realizar la depuración del código en el editor de Python del software RoboDK, enfocándose en la eliminación de espacios en blanco y comentarios generados; la omisión de esta sugerencia no permite la ejecución de la aplicación desarrollada.

Es recomendable cortar la alimentación principal de la celda robotizada cuando esta no se utilice a fin de evitar daños en los componentes de la celda debido a factores externos. Además, se recomienda utilizar el botón STOP de la interfaz UFACTORY Studio para detener el proceso en caso de ser necesario y evitar el uso constante del paro de emergencia físico, con la finalidad de evitar un daño mecánico en el cobot.

Se recomienda que en caso de anomalías en el funcionamiento del cobot UFACTORY Lite 6 o para la actualización de software y firmware; el operario debe contactarse con servicio técnico de la empresa mediante la siguiente dirección: <https://www.ufactory.cc/contact-us/>.

Para una adecuada manipulación del robot y los accesorios de la celda robotizada, se aconseja una lectura previa del manual de operación.

Bibliografía

Albea Sánchez, C. (2004a). *ESTUDIO TÉCNICO Y DE VIABILIDAD DE A PROGRAMACIÓN OFF-LINE DE ROBOTS*. Universidad de Sevilla.

<https://biblus.us.es/bibing/proyectos/abreproy/3721/fichero/PFC.pdf>

Albea Sánchez, C. (2004b). *ESTUDIO TÉCNICO Y DE VIABILIDAD DE LA PROGRAMACIÓN OFF-LINE DE ROBOTS*. UNIVERSIDAD DE SEVILLA.

Aljama, D. A. (2011). *UTILIZACIÓN DE HERRAMIENTAS DE SIMULACIÓN EN LA ROBÓTICA INDUSTRIAL*. Escuela Politécnica Superior Universidad de Lleida.

Amieva, D. (2021). *DISEÑO DE UNA CELDA ROBOTIZADA DE SOLDADURA HÍBRIDA DE ALTA POTENCIA*. Universidad de Oviedo.

Andrade B., K. J., & Curay M., A. J. (2022). *Repotenciación de la celda de soldadura robotizada que incluye el brazo robótico KUKA KR5-Arc, mesa posicionadora, sujetadores y sistema articulado de extracción localizada de humos de soldadura en el Laboratorio de Robótica Industrial de la Universidad de las Fuerzas Armadas ESPE - Sede Latacunga*. [Universidad de las Fuerzas Armadas ESPE-L]. <http://repositorio.espe.edu.ec/xmlui/handle/21000/33784>

Barrientos, A., Peñín, L., Balaguer, C., & Aracil, R. (1997). *Fundamentos de Robótica* (McGRAW-HILL). <https://eltrasteroloco.files.wordpress.com/2017/03/267380685-fundamentos-de-robotica.pdf>

Barrientos, A., Peñín, L. F., Balaguer, C., & Aracil, R. (2005). *Fundamentos de la robótica* (Segunda). McGraw-Hill.

Cecy., T. (2020). *Estructura de un robot industrial*.

<https://www.cecyl3.ipn.mx/estudiantes/plan%20continuidad/Archivo%20comprimido2/morfologia%20de%20un%20robot.pdf>

Celdas robotizadas. (s. f.). Unimate Robótica. Recuperado 15 de mayo de 2023, de <https://unimaterobotica.com/celdas-robotizadas/>

CENTRIC. (2022, marzo 10). ¿Qué son las celdas robóticas y cuáles son sus ventajas? [CENTRIC]. *Distribución y venta de maquinaria CNC*. <https://centricdemexico.com/ventajas-celdas-roboticas/>

CFZ Cobots. (2019). *Robotica industrial y robotica colaborativa | CFZ - Cobots*. <https://cfzcobots.com/robotica-tradicional-y-robotica-colaborativa-diferencias/>

Comíns, F. (2018). *Lenguajes de programación de robots industriales*.

Commare, R. L. (2021). *Trajectory optimization for collaborative robotics applications*. Universidad Politécnica de Milano.

Cuello, F. (2020). *DISEÑO Y SIMULACIÓN DE APLICACIONES ROBOTIZADAS MEDIANTE ROBODK*. Universidad Politécnica de Valencia.

EDS Robotics. (2022, enero 19). ¿Qué es la Robótica Colaborativa? ▷ Aplicaciones y Ventajas. *EDS Robotics*. <https://www.edsrobotics.com/blog/robotica-colaborativa-que-es/>

EmpresaActual. (2022, febrero 2). *Evolución y tendencias de la robótica industrial*. Empresa Actual. <https://www.empresaactual.com/evolucion-y-tendencias-de-la-robotica-industrial/>

Escobar, E. (2022). *Diseño e implementación de un módulo de aprendizaje ciber-físico utilizando herramientas de la industria 4.0 para la simulación y monitoreo de un proceso didáctico en tiempo real para el Laboratorio de Mecatrónica de la universidad de las Fuerzas Armadas "ESPE" sede Latacunga*. Universidad de las Fuerzas Armadas ESPE-L.

Esneca. (2019, noviembre 22). Robótica colaborativa: Qué es y para qué sirve. *Esneca*. <https://www.esneca.com/blog/robotica-colaborativa-que-es/>

Fanuc. (2023). *FANUC - Cobot Brochure*.

Gazebo. (s. f.). *Showcase—Gazebo*. Gazebo. Recuperado 20 de junio de 2023, de <https://gazebosim.org/showcase>

Ginés, M. B. (2023). *DISEÑO DE CELDA ROBÓTICA CON SISTEMA DE ASPIRACIÓN PARA LA MANIPULACIÓN DE PLACAS CERÁMICAS DE GRAN TAMAÑO*. Universidad Jaime I.

Gisginis, A. (2021). *Production line optimization featuring cobots and visual inspection system*.

Gonzalez, F. (2019). *Generación off-line de trayectorias libres de colisiones para manipuladores industriales*. Universidad Nacional de Colombia.

González, V. (2022, marzo). *Robots industriales: Definición y clasificación*.

http://platea.pntic.mec.es/vgonzale/cyr_0204/ctrl_rob/robotica/industrial.htm

Guaraca, P., & Ochoa, J. (2015). *Estudio de la programación y operación de los robots industriales KUKA KR16-2 y KR5-2 ARC HW*. Universidad Politécnica Salesiana Sede Cuenca.

Hader, B. (2021). *Intuitive programming of collaborative human robot processes*.

Hernández, J. C., & Vizán, A. (2015). *Sistemas de automatización y robótica para las pymes españolas*. <http://a.eoi.es/robopyme>

InfoPLC. (2022, octubre 13). *Otro año récord en instalaciones robóticas: Se alcanzan las 3,5 millones de unidades a nivel global - infoPLC*. <https://www.infoplac.net/plus-plus/mercado/item/111847-informe-ifr-instalaciones-roboticas-2021>

Ionescu, T. B. (2020). Leveraging Graphical User Interface Automation for Generic Robot Programming. *Robotics*, 10(1), 3. <https://doi.org/10.3390/robotics10010003>

KICKSTARTER. (2023, febrero 18). *UFACTORY Lite 6 – Most Affordable Collaborative Robot Arm*. Kickstarter. <https://www.kickstarter.com/projects/ufactory/ufactory-lite-6-most-affordable-collaborative-robot-arm>

Kopp, T., Baumgartner, M., & Kinkel, S. (2021). Success factors for introducing industrial human-robot interaction in practice: An empirically driven framework. *The International Journal of Advanced Manufacturing Technology*, 112(3-4), 685-704. <https://doi.org/10.1007/s00170-020-06398-0>

Lange, P. (2020, abril 17). *Estrategias de seguridad para los cobots: Aplicaciones de colaboración seguras y eficaces*. Interempresas. <https://www.interempresas.net/Robotica/Articulos/301888-Estrategias-de-seguridad-para-los-cobots-aplicaciones-de-colaboracion-seguras-y-eficaces.html>

Lascano, C. (2007). *Seguridad en Celdas Robotizadas | PDF | Robot | Robótica*. Scribd. <https://es.scribd.com/document/408634148/Seguridad-en-Celdas-Robotizadas>

Legarreta, J., & Martinez, R. (2020). *TEMA 2 MORFOLOGÍA DEL ROBOT*. https://ocw.ehu.eus/pluginfile.php/50442/mod_resource/content/3/TEMA%202.%20MORFOLOG%20DEL%20ROBOT_OCW_revisado.pdf

Montoya, J. A. Z. (2019). *DISEÑO E IMPLEMENTACIÓN DE UNA CELDA ROBOTIZADA CLASIFICADORA UTILIZANDO UN ROBOT DELTA 3RSS Y UN SISTEMA DE VISIÓN ARTIFICIAL*. UNIVERSIDAD DE PAMPLONA.

Navarro, A. (2020). *Robot industrial. Manual de instalación*. Ediciones Paraninfo, S.A.

NCYT. (2022, agosto 19). *¿Qué es la robótica colaborativa y cómo puede ayudar a las empresas?* [CollectionText]. Noticias de la Ciencia y la Tecnología (Amazings® / NCYT®);

Noticias de la Ciencia. <https://noticiasdelaciencia.com/art/44851/que-es-la-robotica-colaborativa-y-como-puede-ayudar-a-las-empresas>

Odorico, A. H., Lage, F., & Cataldi, Z. (2007). *Interdisciplinariedad aplicada a una tecnología actual, en el diseño de una celda robotizada*.

Razo, A. F. (1999). *ROBIN: Herramienta gráfica de simulación de robots industriales en un ambiente virtual*. Universidad de las Américas Puebla.

Reichelt. (s. f.). *UF VACUUM LITE - Vacuum gripper for Lite 6*. Elektronik Und Technik Bei Reichelt Elektronik Günstig Bestellen. Recuperado 21 de junio de 2023, de <https://www.reichelt.com/de/en/vacuum-gripper-for-lite-6-uf-vacuum-lite-p340112.html>

Reyes Cortés, F. (2011). *Robótica CONTROL DE ROBOTS MANIULADORES* (Primera). Alfaomega.

RoboDK. (2023). *Simulador para robots industriales y programación fuera línea—RoboDK*. <https://robodk.com/es/>

Robotnik. (2022, diciembre 15). ¿Qué es un cobot y cuales son sus beneficios? | Robotnik ®. *Robotnik*. <https://robotnik.eu/es/que-es-un-cobot-y-cuales-son-sus-beneficios/>

Rodríguez, D. (2021). *DISEÑO E IMPLEMENTACIÓN DE UNA CELDA AUTOMATIZADA CON ROBÓTICA COLABORATIVA*. Universidad Politécnica de Valencia.

Salvador, I. (2019). *GUIADO DE UN COBOT CON DISPOSITIVO EXTERNO PARA LA GRABACIÓN DE TRAYECTORIAS*. Universidad Politécnica de Valencia.

Sánchez, Á. (2020). *SISTEMA DE PROGRAMACIÓN FUERA DE LÍNEA PARA ROBOTS DE SOLDADURA BASADO EN UNA INTERFAZ HÁPTICA*. UNIVERSIDAD AUTÓNOMA DE SINALOA.

Senao, V., Fernández, J. A., & Gonzalez, C. (2004). *Seguridad en una célula flexible de producción de piezas de aluminio*. Universidad Politécnica de Cataluña.

Sicma21, P. (2021, octubre 25). *Robots industriales: Qué son, cómo funcionan y sus tipos*. <https://www.sicma21.com/robots-industriales-tecnologia-y-aplicaciones/>

Simumatik. (s. f.). *Use Case: Tknika Challenge*. Simumatik. Recuperado 21 de junio de 2023, de <https://simumatik.com/use-case-tnnika-challenge/>

SIMUMATIK. (2020). *Simumatik—Advanced simulation, digital twin technology*. Simumatik. <https://simumatik.com/>

tecnopl.com. (2016, noviembre 24). Seguridad de cobots: Cuatro tipos de operación colaborativa » tecnopl. *tecnopl*. <https://www.tecnopl.com/seguridad-de-cobots/>

TEGARA Co., Ltd. (2022, julio 16). Desktop-sized 6-DOF robot arm «U Factory Lite 6» | Information dissemination media for R & D TEGAKARI. *TEGAKARI*. <https://www.tegakari.net/en/2022/07/ufactory-lite-6/>

Trelles, C. A. (2021). *DISEÑO Y CONSTRUCCIÓN DE UN PROTOTIPO DE BRAZO ROBÓTICO INDUSTRIAL A ESCALA*. Universidad Nacional de Loja.

Ufactory. (2021a). *UFACTORY Lite 6 Developer Manual*.

Ufactory. (2021b). *UFACTORY Lite 6 User Manual*.

UNIR. (2021). *¿Qué es la robótica industrial? Concepto, usos y futuro*. UNIR. <https://www.unir.net/ingenieria/revista/robotica-industrial/>

Universal Robots. (2023). *Espacio de Trabajo de un Robot | Universal Robots*. <https://www.universal-robots.com/es/blog/espacio-de-trabajo-de-un-robot/>

Valdez, H. (2023). *Tipos de robot*. https://www.academia.edu/35615107/Tipos_de_robot

Vergara, J. C. (2020, enero 20). Seguridad en robots colaborativos (COBOTS). *INTEGRA ingeniería - Aportamos soluciones globales de ingeniería para la industria*. <https://www.integra-sti.com/noticias/seguridad-en-robots-colaborativos-cobots/>

Villani, V., Pini, F., Leali, F., & Secchi, C. (2018). Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications. *Mechatronics*, 55, 248-266. <https://doi.org/10.1016/j.mechatronics.2018.02.009>

Yebra, L. (2022). *EL LENGUAJE V+*. Universidad Politécnica de Cataluña.

Anexos