

UNIVERSIDAD DE ALMERIA

ESCUELA SUPERIOR DE INGENIERÍA

**“Diseño y fabricación de una
herramienta multifunción para un
robot industrial.”**



Curso 2020 / 2021

Alumno/a:

Borja Hierro Garrido

Director/es:

José Luis Torres Moreno

TRABAJO FIN DE MASTER

MASTER EN INGENIERÍA INDUSTRIAL

UNIVERSIDAD DE ALMERÍA



Diseño y fabricación de una herramienta multifunción para un robot industrial.

-Autor-

**Borja Hierro Garrido
ALMERÍA, MAYO 2021**

Director:

Autor:

José Luis Torres Moreno

Borja Hierro Garrido



Diseño y fabricación de una herramienta multifunción para un robot industrial.



AGRADECIMIENTOS

En estas primeras líneas me gustaría mostrar mi agradecimiento a todas aquellas personas que me han acompañado y apoyado en este proceso de aprendizaje y superación a través de mi paso por este máster de Ingeniería Industrial en la Escuela Superior de ingeniería de la Universidad de Almería.

En primer lugar, agradecer el apoyo de mi familia y novia, por su paciencia e inigualable ayuda ofrecida en todo momento. Quisiera también mostrar mi agradecimiento a todos los compañeros y compañeras con los que he compartido estos años de máster, en especial a Antonio y Víctor que llevan conmigo desde que nos conocimos en el grado de Ingeniería mecánica.

Por último, y no por ello menos importante, quisiera agradecer a mi tutor José Luis su implicación en este proyecto, ya que sin su ayuda este trabajo no habría sido posible.



Diseño y fabricación de una herramienta multifunción para un robot industrial.



RESUMEN

En este trabajo se detalla el proceso de diseño y fabricación de una herramienta multifunción orientada a su uso mediante brazo robótico.

En la actualidad existe una clara tendencia hacia la industria 4.0 y la implementación de robots dentro de los procesos industriales. Este TFM se centra en desarrollar una herramienta completamente funcional orientada a ofrecer soluciones para la manipulación de elementos a través del uso de los robots industriales.

A lo largo de este proyecto se seleccionarán las soluciones más satisfactorias para conseguir los objetivos buscados. Se realizarán los diseños y comprobaciones necesarias para garantizar la calidad y el correcto dimensionamiento de la herramienta. Adicionalmente se realizarán las simulaciones y ensayos reales para verificar su correcto funcionamiento dentro del sistema que incluye la unión de la herramienta junto con el robot industrial y su sistema de control.

ABSTRACT

This work provides a detail explanation of the fabrication and design process of a multifunctional tool oriented to its use by means of a robotic arm.

Currently there is a clear trend towards Industry 4.0 and the implementation of robots within industrial processes. This project focuses on developing a fully functional tool aimed at offering solutions for the manipulation of elements through the use of industrial robots.

Throughout this project the most satisfactory solutions will be selected to achieve the desired objectives. The designs and verifications necessary to guarantee the quality and correct sizing of the tool will be done. Additionally, real simulations and tests will be carried out to verify its correct operation within the system that includes the union of the tool together with the industrial robot and its control system.



Diseño y fabricación de una herramienta multifunción para un robot industrial.

DOCUMENTO 1: MEMORIA

1.	INTERESES.....	19
1.1.	MOTIVACION DEL TRABAJO	19
1.2.	LA INDUSTRIA 4.0	20
1.3.	EL DISEÑO CAD-CAM	23
1.4.	OBJETIVOS	25
1.5.	ESTRUCTURA DE LA MEMORIA DEL TFM.	27
2.	OBJETO	29
2.1.	REQUISITOS DE DISEÑO.	29
3.	ESPECIFICACIONES TÉCNICAS.....	31
3.1.	ESTUDIO DE LOS ELEMENTOS INVOLUCRADOS	31
3.1.1.	LOS ROBOTS INDUSTRIALES	31
3.1.2.	ARDUINO	40
3.1.3.	LOS SISTEMAS DE IMPRESIÓN 3D	42
3.1.4.	LOS SERVOMOTORES	46
3.1.5.	SOFTWARE EMPLEADO	47
3.2.	ESPECIFICACIONES LOS DISTINTOS ELEMENTOS INVOLUCRADOS	49
3.3.	ESTUDIO DE LAS POSIBLES SOLUCIONES.....	68
3.3.1.	GEOMETRÍA DEL MECANISMO TIPO PINZA.....	68
3.3.2.	SISTEMA DE CONTROL.....	71
3.4.	SOLUCION SELECCIONADA.....	73
3.4.1.	ELEMENTOS FINALES QUE COMPONEN MI DISEÑO	75
4.	DESARROLLO Y RESULTADOS	77
4.1.	PROCESO DE DISEÑO DE LOS ELEMENTOS MECÁNICOS.....	77
4.1.1.	GEOMETRIA BASICA DEL SISTEMA PINZA-VENTOSA.....	78
4.1.2.	DISEÑO DE LAS PIEZAS FINALES QUE COMPONEN LA HERRAMIENTA.	90
4.1.3.	DISTRIBUCION Y ENSAMBLE DEL CONJUNTO.....	107
4.2.	VERIFICACIÓN DE LAS PIEZAS MEDIANTE SOLIDWORKS.....	113
4.2.1.	SOPORTE SUPERIOR	115
4.2.2.	SOPORTE INFERIOR	115
4.2.3.	CONJUNTO BASE:	115
4.2.4.	RESULTADOS DEL ESTUDIO MEDIANTE SOLIDWORKS:.....	118
4.3.	FABRICACIÓN MEDIANTE IMPRESIÓN 3D	119
4.3.1.	TRANSFERENCIA DE ARCHIVOS	119
4.3.2.	PARÁMETROS CARACTERISTICOS.....	120
4.3.3.	PIEZAS RESULTADO	124



4.4.	HERRAMIENTA FINAL COMPLETA	130
4.5.	PROGRAMACIÓN DE LAS FUNCIONALIDADES DE LA HERRMIENTA.....	136
4.5.1.	ELEMENTOS NECESARIOS.....	137
4.5.2.	ESQUEMA DE FUNCIONAMIENTO CONJUNTO. DIAGRAMA DE CONEXIONES..	138
4.5.3.	PROGRAMACIÓN DE LOS DISTINTOS COMPONENTES	141
4.6.	SIMULACION DEL SISTEMA MEDIANTE ROBOTSTUDIO	156
4.6.1.	CREACIÓN DE LA ESTACIÓN Y EL ENTORNO DE TRABAJO.....	156
4.6.2.	CONFIGURACIÓN DE LA HERRAMIENTA	158
4.6.3.	CREACION DE LAS TRAYECTORIAS.....	160
4.6.4.	CONFIGURACIÓN DEL CONTROLADOR.....	165
4.6.5.	PROGRAMACION DE LAS ACCIONES MEDIANTE LENGUAJE RAPID	167
4.6.6.	SIMULACIÓN DEL SISTEMA.....	172
4.7.	PUESTA EN MARCHA DEL SISTEMA REAL (SISTEMA ROBOT-HERRAMIENTA)	173
4.8.	MEJORAS Y TRABAJOS FUTUROS	176
5.	CONCLUSIONES	177
6.	REVISIÓN BIBLIOGRÁFICA.....	179
ANEXO 1: CÁLCULOS		181
1.	SÍNTESIS DE NÚMERO DE UN MECANISMO DE 4 BARRAS.....	181
2.	SÍNTESIS DIMENSIONAL DEL MECANISMO DE LA PINZA	183
3.	DIMENSIONAMIENTO DEL SISTEMA DE ENGRANAJES.....	185
4.	PUNTOS DE CIERRE PARA LA MANIPULACIÓN DE LAS DISTINTAS PIEZAS	186
5.	ANÁLISIS DE LAS PIEZAS MEDIANTE SOLIDWORKS SIMULATION	189
5.1.	SOPORTE SUPERIOR	189
5.2.	SOPORTE INFERIOR	192
5.3.	CONJUNTO BASE:	194
ANEXO 2: CÓDIGOS		201
FASE 1: ARDUINO NANO 1 + MODULO COMUNICACIÓN (CONJUNTO EMISOR).....		201
FASE 2: ARDUINO NANO (2) + MODULO COMUNICACIÓN.....		203
FASE 3: ARDUINO UNO (1) + SENSOR SHIELD		204
FASE 4: ARDUINO UNO (2)		206
PROGRAMACIÓN IRC5C RAPID.....		207



DOCUMENTO 2: FASES TFM Y CRONOGRAMA

1. FASES Y CRONOGRAMA	3
1.1. FASES DEL TFM	3
1.2. CRONOGRAMA.....	4

DOCUMENTO 3: PLANOS

DOCUMENTO 4: MEDICIONES Y PRESUPUESTO

1. PRESUPUESTO	3
1.1. COMPONENTES DE IMPRESIÓN 3D.....	3
1.2. COMPONENTES COMERCIALES.....	5
1.3. COSTES DE INGENIERÍA	5
1.4. PRESUPUESTO TOTAL.....	6



Diseño y fabricación de una herramienta multifunción para un robot industrial.

INDICE DE FIGURAS

Figura 1. 1. Ejemplo de cadena de producción con brazos robóticos.....	19
Figura 1. 2. Evolución hacia industria 4.0.....	20
Figura 1. 3. Cambios industria 4.0.....	21
Figura 1. 4. Elementos industria 4.0.....	22
Figura 2. 1. Especificaciones de carga IRB 140.....	29
Figura 3. 1. Robot manipulador.....	32
Figura 3. 2. Estructura robot cartesiano [10].....	33
Figura 3. 3. Estructura robot cilíndrico.....	34
Figura 3. 4. Estructura robot polar.....	34
Figura 3. 5. Estructura brazo articulado angular.....	34
Figura 3. 6. Estructura robot SCARA.....	35
Figura 3. 7. Morfología brazo robot.....	35
Figura 3. 8. Manipulador robot.....	36
Figura 3. 9. Partes del manipulador del robot.....	36
Figura 3. 10. Flexpendant ABB.....	37
Figura 3. 11. Brazo robótico ABB.....	38
Figura 3. 12. Ejemplo de volumen de trabajo del robot industrial.....	38
Figura 3. 13. Modelos Arduino.....	41
Figura 3. 14. Primer sistema de impresión 3D.....	42
Figura 3. 15. Impresora mediante inyección.....	43
Figura 3. 16. Impresión mediante Filaflex.....	44
Figura 3. 17. Prótesis fabricada mediante impresión 3D.....	45
Figura 3. 18. Plataforma Thingiverse.....	46
Figura 3. 19. Servomotor desmontado.....	46
Figura 3. 20. Estudio mediante SolidWorks Simulation.....	47
Figura 3. 21. ABB IRB140 de la Universidad de Almería.....	50
Figura 3. 22. Rendimiento ABB IRB 140.....	50
Figura 3. 23. Dimensiones ABB IRB 140.....	51
Figura 3. 24. Área de trabajo IRB140.....	51
Figura 3. 25. Ficha técnica IRB140.....	52
Figura 3. 26. Terminal de acoplamiento muñeca IRB 140.....	53
Figura 3. 27. Controlador ABB IRC5C.....	53
Figura 3. 28. Información técnica IRC5C.....	54
Figura 3. 29. Información técnica Flexpendant.....	54
Figura 3. 30. Impresora 3D BQ WitBox 2.....	56
Figura 3. 31. Servomotor HD-3001HB.....	58
Figura 3. 32. Sistema de ventosas para succión.....	59
Figura 3. 33. Arduino Uno.....	61
Figura 3. 34. Disposición de pines Arduino Uno.....	62
Figura 3. 35. Arduino Nano.....	62

Figura 3. 36. Disposición de pines Arduino Nano.....	63
Figura 3. 37. Arduino sensor Shield V5.0	64
Figura 3. 38. Disposición de pines Sensor Shield v5.0.....	65
Figura 3. 39. Módulo de comunicación NRF24L01.....	66
Figura 3. 40. Pinout nRF24L01.....	66
Figura 3. 41. Pantalla LCD y módulo de comunicación I2C	67
Figura 3. 42. Conexiones del módulo I2C para pantalla LCD.....	67
Figura 3. 43.Sistema pinza mediante cremallera	68
Figura 3. 44. Sistema pinza de doble cremallera	69
Figura 3. 45. Sistema pinza de transmisión directa.....	69
Figura 3. 46.Sistema pinza de transmisión directa con mecanismo de 4 barras	70
Figura 3. 47. Sistema pinza con transmisión indirecta mediante mecanismo de 4 barras.....	70
Figura 4. 1Croquis inicial de la base de la herramienta	79
Figura 4. 2. Mecanismo 4 barras genérico	79
Figura 4. 3.Paralelismo entre elementos de la pinza y eje de la herramienta	80
Figura 4. 4.Croquis de conexiones entre engranajes del mecanismo de 4 barras para la pinza	81
Figura 4. 5.Engranaje de transmisión de potencia desde el servomotor hacia el mecanismo de la pinza	82
Figura 4. 6.Engranaje de transmisión de potencia para la actuación de la pinza	82
Figura 4. 7.Croquis de los puntos de anclaje a la base de los elementos del sistema de 4 barras	84
Figura 4. 8. Acoplamiento de las circunferencias base de ambos engranajes.....	85
Figura 4. 9.Detalle de posición del engranaje y medida eslabón d en el sistema de pinza	85
Figura 4. 10. Dimensión de los eslabones del sistema de 4 barras.....	86
Figura 4. 11. Detalle de paralelismo entre eslabón final y eje de la herramienta	87
Figura 4. 12. Detalle medidas eslabones a. b y f.....	88
Figura 4. 13. Posición de cierre máximo del mecanismo de 4 barras.....	88
Figura 4. 14.Posición de apertura máxima del mecanismo de 4 barras	89
Figura 4. 15. Dimensiones finales mecanismo 4 barras para obtener el sistema de pinza. Ejemplo para una apertura de 60 mm	89
Figura 4. 16. Engranaje de la pieza acoplada al servomotor.....	90
Figura 4. 17.Pieza de acople del servomotor	91
Figura 4. 18. Geometría pieza de acople del servomotor	91
Figura 4. 19.Pieza de acople al servomotor	91
Figura 4. 20. Detalle acople al servomotor	92
Figura 4. 21. Dimensiones finales mecanismo de 4 barras	92
Figura 4. 22. Eslabón con engranaje	93
Figura 4. 23. Eslabón simple.....	93
Figura 4. 24. Eslabón pinza.....	93
Figura 4. 25. Eslabón con engranaje modificado	94
Figura 4. 26. Detalle acoplamiento entre engranajes del sistema pinza	94
Figura 4. 27. Sistema de pinza totalmente cerrado	95
Figura 4. 28. Sistema de pinza totalmente abierto	95
Figura 4. 29. Detalle ensamble del sistema de pinzas.....	96
Figura 4. 30. Resultado del sistema de pinzas. Vista frontal.....	96
Figura 4. 31. Resultado del sistema de pinzas. Vista posterior	97

Figura 4. 32. Base superior y características.	98
Figura 4. 33. Base inferior	98
Figura 4. 34. Detalles base inferior.....	99
Figura 4. 35. Corte base completa por el plano de acople al servomotor	99
Figura 4. 36. Detalle para acople al servomotor	100
Figura 4. 37. Detalle colocación de los engranajes sobre la base	100
Figura 4. 38. Detalle acoplamiento herramienta de ventosas	101
Figura 4. 39. Detalle acoplamiento al robot IRB140	101
Figura 4. 40. Diseño del soporte superior	102
Figura 4. 41, Detalle de altura del soporte superior	103
Figura 4. 42. Soporte superior junto con los elementos a acoplar en el	103
Figura 4. 43. Diseño soporte inferior	104
Figura 4. 44. Detalle altura soporte inferior.....	104
Figura 4. 45. Soporte inferior junto con los elementos a acoplar en el.....	105
Figura 4. 46. Carcasa para sistema de alimentación	105
Figura 4. 47. Diseño soporte para sensor de distancia	106
Figura 4. 48. Detalle de colocación para el soporte del sensor de distancia	106
Figura 4. 49. Vista isométrica izquierda herramienta ensamblada.....	109
Figura 4. 50. Vista isométrica derecha herramienta ensamblada	109
Figura 4. 51. Vistas ensamblaje herramienta.....	110
Figura 4. 52. Interferencia sistema de pinzas con elementos sobre la herramienta.	111
Figura 4. 53. Interferencia entre las pinzas y el soporte superior.....	112
Figura 4. 54. Interferencia entre las pinzas y el soporte inferior	112
Figura 4. 55. PLA Solidworks	114
Figura 4. 56. Simplificación de la herramienta para simulación	116
Figura 4. 57. Posición de estudio y fuerzas aplicadas	116
Figura 4. 58. Mallado de pieza en formato STL.....	119
Figura 4. 59. Parámetros de la impresora dentro de UltimakerCura.....	121
Figura 4. 60. Posicionamiento de las piezas en el entorno UltimakerCura.....	121
Figura 4. 61. Parámetros de impresión 3D (1)	123
Figura 4. 62. Parámetros de impresión 3D (2)	124
Figura 4. 63. Parámetros de impresión 3D (3)	124
Figura 4. 64. Entorno de impresión final.....	125
Figura 4. 65. Detalle de datos aportados por Cura	125
Figura 4. 66. Vista previa de la pieza a imprimir	125
Figura 4. 67. BASE SUPERIOR	126
Figura 4. 68. BASE INFERIOR	127
Figura 4. 69. ESLABONES SIMPLES (6), ESLABONES PINZA (2) Y ESLABONES CON ENGRANAJE (1 Y 2)	127
Figura 4. 70. ACOPLAMIENTO AL SERVOMOTOR.....	128
Figura 4. 71. SOPORTE SUPERIOR	128
Figura 4. 72. SOPORTE INFERIOR	128
Figura 4. 73. CAJA BATERIA	129
Figura 4. 74. SOPORTE PARA SENSOR DE DISTANCIA	129
Figura 4. 75. Ensamblaje soporte superior	130
Figura 4. 76. Ensamblaje soporte inferior	131
Figura 4. 77. Detalle ensamblaje soporte inferior más soporte para alimentación	131
Figura 4. 78. Ensamblaje bases (Vista superior).....	132

Figura 4. 79. Ensamblaje bases (vista inferior).....	132
Figura 4. 80.....	133
Figura 4. 81. Ensamblaje herramienta completa (Vista izquierda)	133
Figura 4. 82. Ensamblaje herramienta completa (Vista derecha).....	134
Figura 4. 83. Detalle conexión sensor de distancia	134
Figura 4. 84. Ensamblaje Sistema Robot-Herramienta completo.....	135
Figura 4. 85. Peso herramienta completa	136
Figura 4. 86. Cableado Arduino Nano- nRF24L01	146
Figura 4. 87. Placa correspondiente a la primera fase de control mediante Arduino	146
Figura 4. 88. Colocación del Arduino Nano dentro del ensamblaje.....	147
Figura 4. 89. Visualización del puerto serie.....	148
Figura 4. 90. Resultados obtenidos de la visualización del puerto Serie	149
Figura 4. 91. Cableado puerto serie Arduino Nano- Arduino Uno.....	149
Figura 4. 92. Situación Arduino Uno sobre la herramienta.....	150
Figura 4. 93. Conexión sistema pantalla y sensor de distancia.....	154
Figura 4. 94. Entorno del Robot en el taller de la UAL.....	157
Figura 4. 95. Localización de los elementos en el entorno de trabajo del robot.....	157
Figura 4. 96. Coordenadas de los distintos elementos del entorno de trabajo del robot	158
Figura 4. 97. TCP herramienta.....	159
Figura 4. 98. Acople herramienta-robot	159
Figura 4. 99. TCP herramienta-robot	160
Figura 4. 100. Puntos creados dentro del entorno RobotStudio	161
Figura 4. 101. Punto referido a la manipulación de piezas tipo grande	162
Figura 4. 102. Punto referido a la manipulación de piezas tipo pequeña	162
Figura 4. 103. Orientación de la herramienta en el punto de trabajo	163
Figura 4. 104. Trayectorias.....	164
Figura 4. 105. Tarjeta DSQC 653 de salidas.....	165
Figura 4. 106. Diseño de la lógica de estación	166
Figura 4. 107. Lógica del gestor de la pinza.....	167
Figura 4. 108. Visualización de trayectorias simuladas.....	172
Figura 4. 109. Funcionamiento sensor distancia.....	173
Figura 4. 110. Ensamble herramienta-robot.....	174
Figura 4. 111. Cableado para unión de los sistemas	175
Figura Anexos 1. Mecanismo de 4 barras implementado.....	181
Figura Anexos 2. Geometría del mecanismo de 4 barras implementado.....	181
Figura Anexos 3. Elementos del mecanismo de 4 barras genérico.....	182
Figura Anexos 4. Ángulos dentro del mecanismo de 4 barras	183
Figura Anexos 5. Dimensiones pieza grande	186
Figura Anexos 6. Ángulo en cierre máximo.....	187
Figura Anexos 7. Ángulos en apertura de 88 mm	187
Figura Anexos 8. Pieza pequeña 44 mm	188
Figura Anexos 9. Ángulo en apertura de 44 mm.....	188
Figura Anexos 10. Fuerzas aplicadas sobre el soporte superior.	189
Figura Anexos 11. Fuerzas generadas por el soporte superior	195
Figura Anexos 12. Cargas correspondientes al soporte inferior	195
Figura Anexos 13. Cargas generadas por el servomotor.....	195

Figura Anexos 14. Cargas generadas por el sistema de ventosas y el sensor de distancia..... 196
Figura Anexos 15. Carga generada por el elemento de manipulación 196

INDICE DE TABLAS

Tabla 1. Especificaciones técnicas IRB140.....	50
Tabla 2. Datos técnicos módulo DSQC 653	55
Tabla 3. Especificaciones BQ Witbox2	57
Tabla 4. Especificaciones servomotor HD-3001HB	58
Tabla 5. Parámetros sistema ventosas.....	59
Tabla 6. Especificaciones técnicas sensor de distancia	60
Tabla 7. Especificaciones Arduino Uno	61
Tabla 8. Especificaciones Arduino Nano	63
Tabla 9. Especificaciones Sensor Shield V5.0	65
Tabla 10. Elementos del sistema.....	75
Tabla 11. Resumen de parámetros de los engranajes	81
Tabla 12. Relación piezas-actuadores	107
Tabla 13. Pieza con cada uno de los elementos que deberá acoplar a ella	108
Tabla 14. Piezas contenidas en el ensamblaje de la herramienta	110
Tabla 15. Desarrollo de componentes que intervendrán dentro del estudio de SolidWorks Simulation	113
Tabla 16. Resultados estudio soporte superior	115
Tabla 17. Resultados estudio soporte inferior	115
Tabla 18. Resultados estudio base 1kg	117
Tabla 19. Resultados estudio base 2kg	117
Tabla 20. Resultados estudio base 0,5kg	118
Tabla 21. Piezas, tiempos y pesos de impresión 3D.....	126
Tabla 22. Elementos necesarios para la programación de funcionalidades	137
Tabla 23. Correlación puntos creados con la posición correspondiente	161
Tabla 24. Configuración de los eslabones	186
Tabla 25. Fuerzas sobre el soporte superior	189
Tabla 26. Mallado soporte superior	190
Tabla 27. Tensiones soporte superior	190
Tabla 28. Desplazamientos soporte superior.....	191
Tabla 29. Fuerzas sobre el soporte inferior.....	192
Tabla 30. Mallado soporte inferior	192
Tabla 31. Tensiones soporte inferior	193
Tabla 32. Desplazamientos soporte inferior	193
Tabla 33. Sujeciones entre componentes del modelo de la base.....	194
Tabla 34. Contactos aplicados en el modelo de la base.....	194
Tabla 35. Fuerzas aplicadas al conjunto de la base.....	194
Tabla 36. Tensiones en el conjunto base para manipulación de 1Kg	197
Tabla 37. Desplazamientos en el conjunto base para manipulación de 1Kg	197
Tabla 38. Tensiones en el conjunto base para manipulación de 2Kg	198
Tabla 39. Desplazamientos en el conjunto base para manipulación de 2Kg	198



Tabla 40. Tensiones en el conjunto base para manipulación de 0.5Kg	199
Tabla 41. Desplazamientos en el conjunto base para manipulación de 0.5Kg	199

DOCUMENTO 4: PRESUPUESTO

Tabla 1. Tiempos y peso componentes 3D.....	4
Tabla 2. Importe piezas individuales.....	4
Tabla 3. Costes asociados impresión 3D	4
Tabla 4. Importes componentes comerciales	5
Tabla 5. Costes de ingeniería	5
Tabla 6. Presupuesto de ejecución	6
Tabla 7. Presupuesto final.....	6

INDICE DE ESQUEMAS

Esquema 1. Diagrama de conexiones entre sistemas Arduino	73
Esquema 2. Diagrama de conexiones del sistema de control.....	139
Esquema 3. Diagrama de conexiones del sistema sensor de distancia-pantalla	140
Esquema 4. Esquema general de salidas/Entradas para el funcionamiento del ciclo de control	141
Esquema 5. Ciclo de acción según la salida del controlador activa	165



DOCUMENTO1: MEMORIA.



Diseño y fabricación de una herramienta multifunción para un robot industrial.

1. INTERESES

1.1. MOTIVACION DEL TRABAJO

En la actualidad, el uso de los brazos robóticos ya está ampliamente instaurando dentro de gran cantidad de procesos que engloban la industria de la ingeniería. Desde manipulación de cargas hasta procesos de fabricación automatizados, el uso de estos sistemas se ha destacado como una de las soluciones de mayor éxito.

Claros ejemplos de ello son las fábricas automovilísticas, donde se integran los brazos robóticos para formar cadenas de producción totalmente automatizadas [Figura 1.1].



Figura 1. 1. Ejemplo de cadena de producción con brazos robóticos

La evolución hacia la llamada industria 4.0 de la que se hablará más adelante en este TFM, también hace necesaria la implementación de nuevas herramientas capaces de adaptarse a las tecnologías existentes de cara a conseguir una industria con un mayor grado de productividad y eficiencia. Dentro de la nueva generación de herramientas capacitadas para hacer evolucionar a la industria actual se encuentran los mencionados brazos robóticos.

Es por todo ello, por lo que la gama de herramientas orientadas a estos robots y las funciones que son capaces de desarrollar son cada día más amplias. Si bien es cierto que las herramientas más sofisticadas para estos brazos robóticos tienen un precio muy elevado y no siempre cumplen los requisitos para los cuales se pretende usar dicha herramienta.

Otra tecnología con gran auge en los últimos años dentro de la industria es la impresión 3D, mediante la cual se pueden obtener prototipos y piezas en distintos materiales, con relativa rapidez y bajo coste. Esta tecnología es de gran interés dentro de este TFM ya que proporciona la capacidad de, a través de diseño CAD-CAM, poder diseñar y fabricar nuestros propios modelos.

Gracias a los medios proporcionados por la Universidad de Almería, dentro de las instalaciones del CITE IV, podemos contar tanto con un brazo robótico industrial de la marca ABB como con una impresora 3D. Es por ello por lo que se ha decidido realizar un proyecto con el fin de diseñar y fabricar una herramienta para este brazo robótico haciendo uso de las tecnologías disponibles. A fin de proporcionar a esta herramienta una mayor proyección dentro de un mercado que avanza hacia un alto grado de automatización, se pretende dotar al sistema de la capacidad de operar de manera autónoma.

1.2. LA INDUSTRIA 4.0

Cuando se hace referencia al concepto de industria 4.0, este está referido a la inclusión de las distintas tecnologías digitales dentro de las fábricas y el entorno industrial. El término Industria 4.0 hace referencia a la denominada cuarta revolución industrial que supone la transformación digital de la industria mediante la integración y la digitalización de todos los procesos industriales que forman parte de la cadena de valor. Se caracteriza por una adaptabilidad, flexibilidad y eficiencia que permite cubrir las necesidades de los clientes en el mercado actual.

Expresado de una forma sencilla, la industria 4.0 se refiere a la digitalización de los procesos productivos dentro del entorno industrial. Esta digitalización se lleva a cabo mediante sensores y sistemas de información que interactúan con los procesos productivos y los transforman haciéndolos más eficientes.

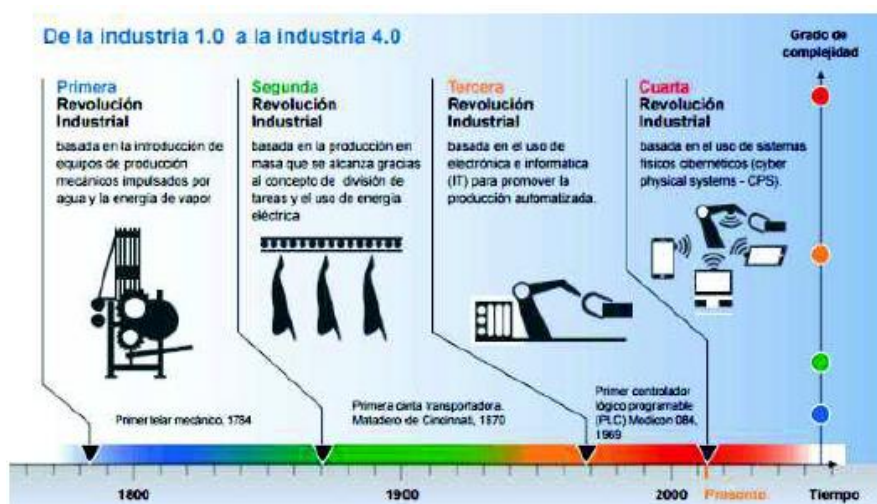


Figura 1. 2.Evolución hacia industria 4.0

Esta evolución hacia la industria 4.0 [Figura 1.2.] supone una modificación en la mentalidad clásica que se tiene actualmente de la fábrica, ya que la inclusión de los distintos sistemas y sensores dentro de ella se traducirá en una mayor competitividad, pues afecta de manera directa sobre los costes de mano de obra, los niveles de compromiso social y los niveles de energía utilizados.

Algunas de las ventajas más significantes que ofrece la industria 4.0 a sus beneficiarios son:

- Una mayor capacidad para la adaptación continua a los cambios en la demanda.
- Reducción del porcentaje de defectos en fábrica.
- Mayor posibilidad de personalizar el servicio al cliente.
- Implantación de un servicio postventa.
- Diseñar, producir y vender productos en un menor margen de tiempo.
- Disminuir los tiempos en las series de producción haciéndolas más rentables.
- Capacidad de análisis de los productos desde múltiples canales.

¿QUÉ CAMBIOS SUPONEN EN LA EMPRESA CON LA INDUSTRIA 4.0?

INDUSTRIA TRADICIONAL	INDUSTRIA 4.0
Producción en masa	Personalización según los requerimientos del cliente
Grandes factorías para fabricar grandes volúmenes de un producto en específico	Factorías inteligentes con una producción flexible a un coste competitivo
Planificación ordenada basada en una previsión con un stock	Producción dinámica según la demanda
Compra del producto	Uso del producto como servicio
Minimización de los costes	Maximización ROCE: rentabilidad/capital utilizado
Rigidez laboral	Flexibilidad en la organización de trabajo

Figura 1. 3. Cambios industria 4.0

La industria 4.0 [Figura 1.3] aporta una mayor flexibilidad dentro del proceso productivo ya que agrega la inteligencia a los sistemas y herramientas de la fábrica, tales como control de existencias, estado de los subproductos, depósitos, etc. De esta forma, la comunicación a través de la red interna de la fábrica mejora y permite mayor adaptabilidad frente a situaciones inesperadas. Todo ello se verá traducido en un aumento y mejora de la producción.

Dentro de este nuevo modelo de industria los puntos de mayor relevancia serán:

- **Interoperabilidad:** La capacidad de los distintos sistemas dentro de la fábrica de comunicarse entre ellos.
- **Descentralización:** La capacidad de implementar elementos ciber físicos con capacidad de tomar decisiones de forma autónoma.
- **Análisis en tiempo real:** Posibilidad de recopilar y analizar una gran cantidad de datos provenientes de los sistemas de la fábrica (BIG DATA), de manera que se facilite el proceso de monitorización y subsanación de errores en tiempo real.
- **Virtualización:** La capacidad que se consigue de obtener modelos y simulaciones de la fábrica. También se podrá recopilar datos para obtener una “copia virtual” de la fábrica.
- **Orientación al servicio:** Un proceso con mayor orientación a hacer llegar el valor generado al cliente como nuevos servicios o servicios mejorados
- **Modularidad y escalabilidad:** Una mayor flexibilidad para adaptarse en los cambios de demanda de la industria.



Figura 1. 4. Elementos industria 4.0

Dentro de este desarrollo, el reto al que se enfrenta la industria es en como liderar el proceso de transformación digital adaptando a las personas en su nueva labor, en entornos conectados entre sí [Figura 1.4].

Teniendo en mente todas estas características, se puede afirmar que las nuevas herramientas que se implementan en la industria han de estar orientadas a cumplir con todos los puntos expuestos anteriormente. De esta manera lo que se consigue es una fábrica que economiza energía, tiempos y materias primas.

1.3. EL DISEÑO CAD-CAM

El término CAD-CAM proviene de sus siglas en inglés “Computer-Aided Design” para el término CAD y “computer-Aided Manufacturing” para CAM. Su traducción al español es la de “Diseño asistido por computadora” y “fabricación asistida por computadora”.

Esta disciplina abarca distintos campos desde el diseño gráfico hasta el manejo de bases de datos para el diseño y fabricación. También incluirá el control numérico de las máquinas herramientas.

Esta tecnología tiene su origen en los años 70 dentro de la industria automovilística y de la aviación. Esto proporcionaba una gran ventaja competitiva que permitió a las marcas de este sector una gran diferenciación y apostar por nuevos desarrollos en la tecnología.

Posteriormente el uso de esta tecnología se extendió a otro tipo de sectores e industrias, lo cual permitió un gran avance en los procesos tecnológicos.

El proceso de CAD-CAM consiste, básicamente, en diseñar a través de un software de ordenador una pieza, por ejemplo, darle los parámetros necesarios y realizar el proceso de fabricación asistida, de la que también se encarga el ordenador.

Los pasos necesarios para la producción a través de CAD CAM se pueden resumir en los siguientes puntos:

1. Diseño de la pieza en 3D a través software específicos para ello como SolidWorks.
2. Procesado por parte del programa CAM de los elementos y formas que se van a mecanizar.
3. Diseño de la estrategia de mecanizado de las operaciones necesarias.
4. Elección de los parámetros característicos del proceso de fabricación como herramientas, velocidades y avances que se van a aplicar.
5. Simulación de trayectorias.
6. Postprocesar el programa a un lenguaje accesible por la máquina.

7. Envío del programa.
8. Ejecución del mecanizado.
9. Verificación de la pieza.
10. Preparación para la entrega al cliente o su uso dentro de la industria.

Herramientas como la mencionada SolidWorks te permiten realizar la mayor parte de este proceso de una manera fácil y ágil.

Los usos más comunes para el sistema CAD son:

1. Obtención de planos y mayor uniformidad entre ellos.
2. Obtención de animaciones, simulaciones y análisis cinemáticos de las distintas piezas.
3. Crear modelos 3D para una mayor claridad.
4. Traslado de datos a otros programas para obtener presentaciones, informes y cálculos.

Los usos más comunes para el sistema CAM son:

1. Fabricación y diseño de las piezas.
2. Uso de la base de datos para la fabricación y el diseño de piezas.
3. Calendarización para robots industriales, control numérico y computarizado.
4. Inspección y control de calidad.

Los principales beneficios del sistema CAD CAM son:

- Aumento de la productividad
- Reducción de costes de desarrollo del producto
- Aumento de la calidad y disminución de error en los productos
- Flujos de trabajo rápidos y que facilitan el mecanizado de los productos.
- Gestión precisa de los procesos

1.4. OBJETIVOS

Como se ha indicado en el punto anterior a cerca de la motivación de este trabajo, para este TFM se tiene acceso a las tecnologías de impresión 3D y el brazo robótico existente en las instalaciones de la universidad de Almería. Es por ello por lo que se pretende llevar a cabo el diseño y fabricación de una herramienta multifuncional para este robot industrial.

El objetivo principal de este trabajo es el diseño y fabricación de una herramienta multifunción para acoplarla al brazo robótico ABB IRB 140. Esta herramienta contará con dos modos de manipulación de objetos: mediante agarre con un mecanismo tipo pinza y mediante sujeción con ventosas, es por ello por lo que a lo largo del desarrollo de este proyecto nos referiremos en muchas ocasiones a ella como herramienta pinza-ventosa. Se le dotará de un control electrónico que involucrará el uso del controlador del robot ABB y etapas controladas mediante microcontroladores (placas Arduino). Este proceso de control incluirá una fase de automatización por la cual el conjunto herramienta-robot pueda realizar la manipulación de objetos de forma autónoma siguiendo el ciclo establecido por el operario del sistema. Se le incluirá también un sensor de distancia junto con una pantalla LCD para mostrar la proximidad a objetos desde el frontal de la herramienta. Tras su diseño se procederá a su fabricación mediante tecnología de impresión 3D. Una vez fabricada, se acoplarán todos los sistemas adicionales y se procederá a la implementación real de la herramienta.

Para ello se deberán alcanzar los siguientes subobjetivos:

- **Dimensionamiento en función de los requerimientos de uso:** teniendo en cuenta las herramientas auxiliares de las que se dispone (servomotor, placas Arduino, bomba-ventosas de succión y sensor de distancia) y las tareas a realizar por la herramienta, se alcanzará un dimensionamiento previo que satisfaga estas necesidades.
- **Diseño y ensamblaje:** una vez establecidas las dimensiones de cada elemento se llevará a cabo el diseño y ensamblaje de éstos mediante el uso de herramientas como SolidWorks. En este diseño se tendrán en cuenta las posibles interferencias y posiciones a alcanzar por la herramienta.
- **Comprobación del cumplimiento de criterios de funcionalidad y resistencia mecánica:** una vez dimensionados y diseñados los elementos de la herramienta será necesario comprobar a su correcta resistencia ante las cargas. Esta tarea se realizará a través de las herramientas Simulation proporcionada por SolidWorks.
- **Construcción de las distintas partes de la herramienta y ensamblaje.** A través de impresión 3D se fabricarán las distintas piezas que componen esta

herramienta y se montará el conjunto añadiendo los sistemas externos necesarios.

– **Programación de las funcionalidades de la herramienta:** se programará esta herramienta para que sea capaz de manejar objetos a través de dos formas de sujeción distintas: mediante agarre mecánico con pinza o mediante sujeción por las ventosas acopladas en su extremo. Este proceso incluirá una fase de automatización de las funciones realizadas por el conjunto robot-herramienta a través del propio controlador del robot y los microcontroladores incluidos en el diseño de la herramienta.

– **Se realizará un programa tipo** a través del software robotstudio y mediante la programación de Arduino y del controlador IRC5C del robot para la ejecución de las actuaciones del sistema robot-herramienta. Se contará con el sensor de distancia para una mejora de las capacidades del sistema.

– **Puesta en marcha de la herramienta:** se realizará una prueba de funcionamiento real haciendo uso de todas las herramientas mencionadas previamente. Esta prueba consistirá en la manipulación de una serie de cargas por parte del sistema completo (brazo robótico existente en el CITE IV de la universidad de Almería y herramienta diseñada) de forma totalmente autónoma.

1.5. ESTRUCTURA DE LA MEMORIA DEL TFM.

La presente memoria se ha estructurado en 6 apartados cuyo contenido se resume a continuación.

En este primer apartado se han expuesto las motivaciones que han llevado a la elaboración de este proyecto, con una introducción a los conceptos de industria 4.0 y diseño CAD CAM que serán de vital importancia para comprender estas motivaciones. Posteriormente se han desarrollado los objetivos que se pretenden conseguir con la realización del proyecto correspondiente.

Dentro del segundo apartado se ha incluido el objeto del TFM, donde se establecen de manera clara los requisitos de diseño del sistema final a alcanzar.

En el apartado 3 se muestran las especificaciones técnicas de los elementos que compondrán el sistema y se establecen los conceptos necesarios para poder realizar el diseño y la simulación de esta. Tras ello se muestra el diseño objetivo que se perseguirá con la realización del cuarto punto.

El capítulo 4 de desarrollo y resultados muestra los pasos que se han dado para llegar al diseño final de la herramienta, incluyendo su control, fabricación, simulación y puesta en funcionamiento. Para ello se ha desarrollado cada una de las partes del proyecto desde el dimensionamiento de todos los elementos hasta su diseño final y verificación.

Dentro del apartado 5 se muestra las conclusiones del presente trabajo.

Por último, en el apartado 6 se realiza la revisión bibliográfica correspondiente.



Diseño y fabricación de una herramienta multifunción para un robot industrial.

2. OBJETO

2.1. REQUISITOS DE DISEÑO.

En este punto se enumeran los distintos requisitos que se han de tener en cuenta a la hora de realizar el diseño de nuestro sistema. Todo ello viene dado por los elementos con los que se han de trabajar, así como la utilidad que se le desea dar a la herramienta.

- La herramienta estará diseñada para su posterior adaptación al brazo robótico ABB IRB 140 presente en las instalaciones de la Universidad de Almería.
- El peso máximo del conjunto herramienta-actuadores y carga ha de ser de 6kg como indica la ficha técnica del robot IRB 140 [figura 2.1].

Specification				
Robot version	Reach of 5th axis (m)	Handling capacity (kg)	Supplementary load: - on upper arm (kg)	- on wrist (kg)
IRB 140/ IRB 140T	0.8	6	1	0.5

Figura 2. 1.Especificaciones de carga IRB 140

- La configuración de la herramienta debe permitir la manipulación de piezas mediante dos sistemas distintos: sistema de pinzas y sistema de agarre por ventosas. Este requerimiento implica que la herramienta ha de tener la capacidad de manipular objetos cuya geometría plana imposibilite a la pinza su correcta sujeción. Es por ello por lo que, mediante el método de la ventosa, se obtendrá un agarre correcto de piezas con geometrías planas como discos. Para una correcta convivencia de ambos métodos de sujeción ha de tenerse en cuenta la no interferencia entre ellos a la hora del diseño.
- El sistema de agarre mediante pinza ha de tener capacidad para la manipulación de piezas de un tamaño hasta 200 mm de lado. Esto implica una apertura correcta de la pinza y un alcance apropiado de la misma de al menos la mitad del lado máximo (100 mm).
- La capacidad de carga máxima vendrá determinada por las características de las distintas herramientas a disposición de este TFM (servomotor y herramienta de agarre por ventosas) así como la resistencia a los esfuerzos de las piezas diseñadas. La carga máxima del conjunto de la herramienta y las piezas a manipular nunca debe superar los 6kg ya que esta es la carga máxima del brazo robot.
- Los elementos que componen la herramienta deben ser diseñados para su posterior fabricación en PLA mediante impresión 3D. Esto supone un proceso

de diseño que no involucre geometrías las cuales pudiesen ser problemáticas para su fabricación.

- Se diseñará la herramienta para que a través del controlador IRC5C del propio robot y un conjunto de microcontroladores (placas Arduino o similares) sea capaz de ejecutar las instrucciones de manipulación de elementos. Se proporcionará así al sistema la capacidad de operar de forma autónoma o mediante un sistema de control programable por un operario.
- El sistema ha de ser diseñado de manera que trabaje haciendo uso de las herramientas que se ponen a disposición por parte del tutor del TFM. Se detallarán las herramientas y sus modelos de manera más exhaustiva en puntos posteriores de este TFM, en este punto se enumerarán a modo de exposición inicial:
 - Servomotor. *Modelo Power HD - Servomotor Standard - HD-3001HB*
 - Sistema de ventosas con bomba de succión. [Kit para 25T Servos MG996 MG995 DS3218]
 - Sensor de distancia VL53L0X.
 - Se ponen a disposición diferentes modelos de placas Arduino junto con sistemas complementarios para su conectividad. Estos sistemas se detallarán en puntos posteriores.

La existencia de sistemas preestablecidos a los que debe adaptarse nuestro diseño implicará un proceso de ingeniería inversa para diseñar una herramienta que se adapte satisfactoriamente a todos estos sistemas y haga funcionar el conjunto de manera que se cumplan todas las especificaciones.

3. ESPECIFICACIONES TÉCNICAS

3.1. ESTUDIO DE LOS ELEMENTOS INVOLUCRADOS

Dentro de este subapartado se realizará un estudio previo de los elementos más característicos que formarán parte del proyecto. Estos elementos serán:

- **Los robots industriales:** como ya se ha puesto de manifiesto, la herramienta en la que está centrado este proyecto estará diseñada para su uso mediante el acople a un brazo robótico. Es por ello por lo que se verán las características y elementos de los sistemas robóticos industriales.
- **Sistema Arduino:** Elemento de vital importancia para dotar a nuestro proyecto de un sistema de control.
- **Los sistemas de impresión 3D:** se estudiará la impresión 3D como método por el cual se han conseguido fabricar los distintos componentes diseñados que formarán parte de la herramienta.
- **Software involucrado en el desarrollo del proyecto.** Se estudiarán los 4 software principales que intervienen dentro del desarrollo del proyecto.
- **Los servomotores.** El uso de servomotores será vital para conseguir las funcionalidades requeridas dentro de nuestro sistema.

3.1.1. LOS ROBOTS INDUSTRIALES

3.1.1.1. DEFINICION DE ROBOT INDUSTRIAL

A la hora de definir qué es un robot industrial existen ciertas dificultades ya que, dependiendo del mercado, existen diferencias entre las distintas definiciones. [3] Para el mercado japonés un robot industrial se define como cualquier dispositivo mecánico con articulaciones móviles destinado a la manipulación. El mercado occidental es más restrictivo ya que exige mayor complejidad, sobre todo en lo relativo al control. Es por ello por lo que en este último mercado podemos distinguir con mayor facilidad entre un robot industrial y un manipulador.

Según el organismo que lo define un robot industrial puede tener distintas definiciones:

- Según la Asociación de industrias robóticas (RIA) Un robot industrial es *“un manipulador multifuncional reprogramable, capaz de mover materias, piezas, herramientas, o dispositivos especiales, según trayectorias variables, programadas para realizar tareas diversas”*.

- La Organización Internacional de Estándares (ISO) con la que se familiariza más el mercado español define robot industrial como *“un manipulador multifuncional reprogramable con varios grados de libertad, capaz de manipular materias, piezas, herramientas o dispositivos especiales según trayectorias variables programadas para realizar tareas diversas”*.
- La Asociación Francesa de Normalización (AFNOR) trata de ser un poco más exacta a través de la definición por separado de Manipulador y Robot:
 - Manipulador: *“mecanismo formado generalmente por elementos en serie, articulados entre sí, destinado al agarre y desplazamiento de objetos. Es multifuncional y puede ser gobernado directamente por un operador humano o mediante dispositivo lógico”*.
 - Robot: *“manipulador automático servo-controlado, reprogramable, polivalente, capaz de posicionar y orientar piezas, útiles o dispositivos especiales, siguiendo trayectorias variables reprogramables, para la ejecución de tareas variadas. Normalmente tiene la forma de uno o varios brazos terminados en una muñeca. Su unidad de control incluye un dispositivo de memoria y ocasionalmente de percepción del entorno. Normalmente su uso es el de realizar una tarea de manera cíclica, pudiéndose adaptar a otra sin cambios permanentes en su material”*.

3.1.1.2. CLASIFICACION DE LOS ROBOTS INDUSTRIALES.

Según su definición también hará distintas formas de clasificación dentro de los robots industriales, basándose en sus características, diseño y especificaciones:

- Manipuladores: Sistemas mecánicos multifuncionales los cuales están controlados mediante sistemas sencillos para gobernar sus elementos. Pueden ser manuales (operario lo controla directamente), De secuencia fija (repetición constante del proceso) o de secuencia variable. Estos sistemas se utilizan para tareas sencillas y repetitivas. [figura 3.1].



Figura 3. 1. Robot manipulador

- Robots de repetición o aprendizaje: Referidos a robots del tipo manipuladores que se encuentran limitados a la repetición de una secuencia de movimientos.
- Robots con control por computador: Son manipuladores controlados por un computador, que habitualmente suele ser un microordenador. Se controlan mediante un lenguaje específico compuesto por instrucciones que se le dan al robot a través de softwares informáticos.
- Robots inteligentes: similares a los del grupo anterior, pero con la capacidad de relacionarse con el mundo que les rodea mediante sensores. Pueden tomar decisiones en tiempo real (autoprogramable).

La clasificación más relevante dentro del desarrollo de este TFM será su clasificación según el tipo de configuración morfológica. Principalmente son cuatro las estructuras clásicas en los manipuladores y están relacionadas con los correspondientes modelos de coordenadas en el espacio y que se citan a continuación: cartesianas, cilíndricas, esféricas, angulares. De igual manera, el brazo del manipulador podrá presentar cuatro configuraciones clásicas:

- **Cartesiana / Rectilínea:** [figura 3.2] El posicionado se lleva a cabo en el espacio de trabajo con las articulaciones prismáticas. Tiene tres grados de libertad correspondientes a tres movimientos lineales (movimientos en los ejes X, Y y Z).

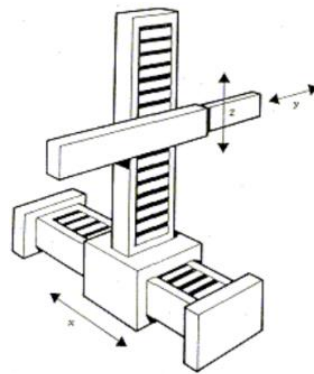


Figura 3. 2. Estructura robot cartesiano [10]

- **Cilíndrica:** [figura 3.3] El robot está dotado de un movimiento de rotación sobre una base, una articulación prismática para la altura y otra prismática para el radio. Es capaz de realizar dos movimientos lineales y uno de rotación, por lo que presenta tres grados de libertad.

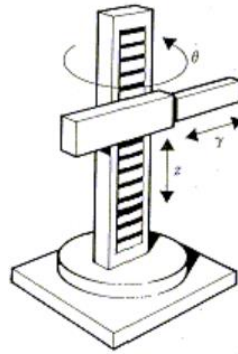


Figura 3. 3. Estructura robot cilíndrico

- **Esférica / Polar:** [figura 3.4] Formado por dos juntas de rotación y una prismática que permiten al robot apuntar gran cantidad de direcciones distintas. Tendrá además la capacidad de extender la mano a una pequeña distancia en sentido radial. Los movimientos que es capaz de realizar son: rotacional, angular y lineal.

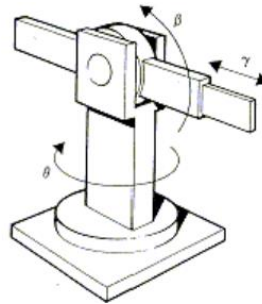


Figura 3. 4. Estructura robot polar

- **De brazo articulado/Angular:** [figura 3.5] Para su posicionamiento este robot usa 3 juntas de rotación. El volumen de trabajo es esférico. Estos tipos de robot tienen grandes similitudes con el brazo humano ya que están compuestos por una cintura, el hombro, el codo y la muñeca. Presentan una articulación con movimiento rotacional y dos con movimiento angular.

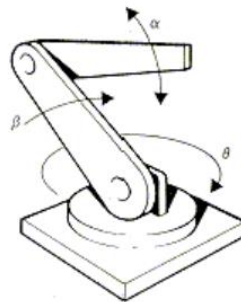


Figura 3. 5. Estructura brazo articulado angular

- **SCARA:** [figura 3.6] De similar configuración con el robot de estructura cilíndrica, pero en este modelo el radio y la rotación se obtienen mediante uno o dos eslabones. Este brazo puede realizar movimientos horizontales de mayor alcance debido a sus dos articulaciones rotacionales. El robot de configuración SCARA también puede hacer un movimiento lineal mediante su tercera articulación.

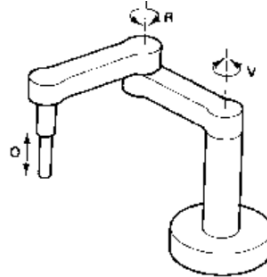


Figura 3. 6. Estructura robot SCARA

3.1.1.3. ROBOT INDUSTRIAL DE ESTUDIO DENTRO DE ESTE PROYECTO

Dentro de este proyecto nos centraremos en el robot del tipo brazo robótico industrial con configuración de Brazo articulado/Angular, puesto que es de esta tipología el que se utilizará en su desarrollo.

Estos robots cuentan con una estructura cuyos principales elementos son los siguientes:

- Estructura mecánica.
- Transmisiones.
- Actuadores.
- Sensores.
- Elementos terminales.
- Controlador.

La construcción física de la mayor parte de los robots industriales tiene un alto grado de similitud con la morfología propia del cuerpo humano, es por ello por lo que sus elementos suelen nombrarse haciendo referencias a partes del cuerpo [figura 3.7]:

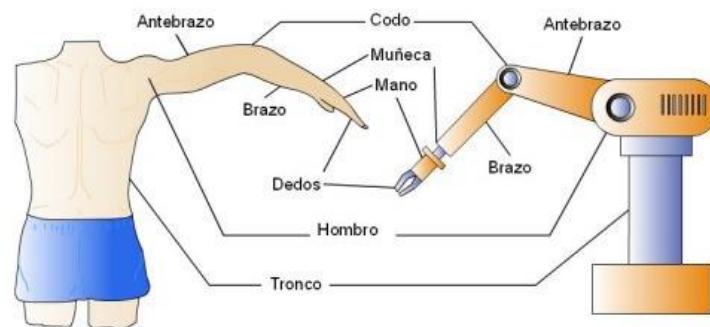


Figura 3. 7. Morfología brazo robot

Dentro de estos elementos que formarán parte del robot, cabe destacar las principales características de los siguientes:

- **Manipulador:** Componente principal formado por elementos estructurales unidos mediante articulaciones. Esto permitirá un movimiento relativo entre cada uno de sus eslabones. [figura 3.8]

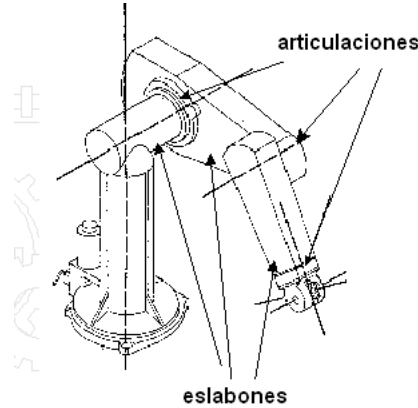


Figura 3. 8. Manipulador robot

Las partes que componen el cuerpo del manipulador por lo general serán las siguientes [figura 3.9]:



Figura 3. 9. Partes del manipulador del robot

- **Controlador:** Es el elemento que regula cada uno de los movimientos del brazo robótico. Se encarga de procesar la información y transmitir las señales correspondientes para una correcta actuación del brazo robot.
- **Dispositivos de entrada y salida:** Son los dispositivos encargados de transmitir las ordenes de entrada para que actúe el robot. Un claro ejemplo es el denominado Flexpendant de la marca ABB [figura 3.10], mediante el cual puede controlarse el robot de manera directa por un operario.



Figura 3. 10. Flexpendant ABB

Los dispositivos de entrada y salida permiten introducir y ver los datos del controlador. Sirven para mandar instrucciones al controlador y para dar de alta programas de control.

Las señales de entrada y salida se obtienen mediante tarjetas electrónicas instaladas en el controlador del robot. Estas tarjetas también le permiten tener comunicación con otras máquinas-herramientas

3.1.1.4. PRINCIPALES CARACTERÍSTICAS DE LOS ROBOTS INDUSTRIALES

Las características más relevantes dentro de los distintos robots que se pueden encontrar en el mercado son las siguientes:

- **Grados de libertad:** Parámetros necesarios para determinar la posición y la orientación del elemento terminal del manipulador. Cada uno de los movimientos independientes (giros y desplazamientos) que puede realizar cada articulación con respecto a la anterior. El número de grados de libertad del robot viene definido por la suma de los grados de libertad de las articulaciones que lo componen. Puesto que las articulaciones empleadas suelen ser únicamente de rotación y prismáticas, con un solo grado de libertad cada una, el número de GDL del robot suele coincidir con el número de articulaciones que lo componen.

Ya que para posicionar y orientar un cuerpo de cualquier manera en el espacio son necesarios seis parámetros, tres para definir la posición y tres para la orientación, si se pretende que un robot posicione y oriente su extremo (y con él la pieza o herramienta manipulada) de cualquier modo en el espacio, se precisará al menos seis grados de libertad.

En la imagen [figura 3.11] se puede observar un brazo robótico con 6 grados de libertad:



Figura 3. 11. Brazo robótico ABB

- Espacio (volumen) de trabajo:** El volumen de trabajo de un robot se refiere únicamente al espacio dentro del cual puede desplazarse el extremo de su muñeca. Para determinar el volumen de trabajo no se toma en cuenta el actuador final. La razón de ello es que a la muñeca del robot se le pueden adaptar herramientas de distintos tamaños. Para determinar el volumen de trabajo de un robot industrial, el fabricante generalmente indica un plano con los límites de movimiento que tiene cada una de las articulaciones del robot. [figura 3.12]

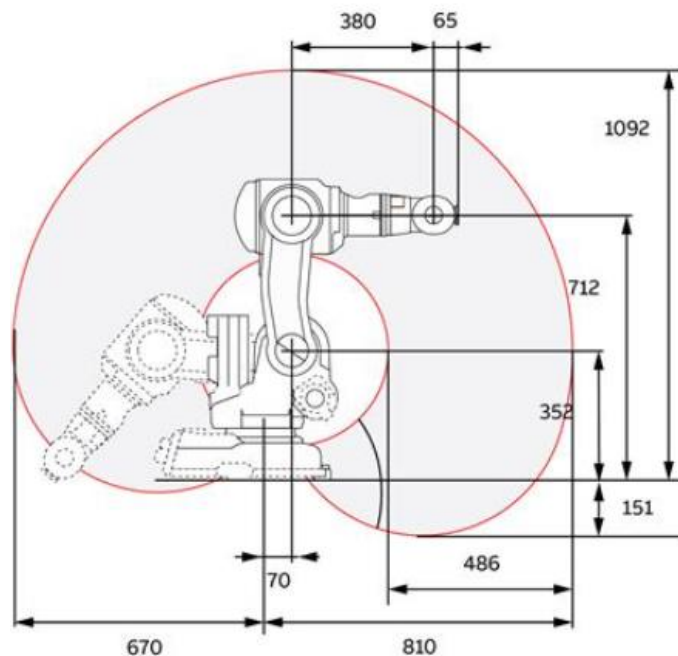


Figura 3. 12. Ejemplo de volumen de trabajo del robot industrial

- **Precisión de los movimientos:** La precisión de movimiento en un robot industrial depende de tres factores:
 - Resolución espacial: incremento más pequeño de movimiento en que el robot puede dividir su volumen de trabajo.
 - Exactitud: capacidad de un robot para situar el extremo de su muñeca en un punto señalado dentro del volumen de trabajo. Mide la distancia entre la posición especificada, y la posición real del robot.
 - Repetitividad: capacidad del robot de regresar al punto programado las veces que sean necesarias. Esta magnitud establece el grado de exactitud en la repetición de los movimientos de un manipulador al realizar una tarea programada.

- **Capacidad de carga:** hace referencia al peso en kg que puede transportar el efector final del brazo robótico. Este dato es proporcionado por los fabricantes.

- **Velocidad:** Se refiere a la velocidad máxima alcanzable por el punto de trabajo final de la muñeca o por las articulaciones.

- **Tipo de actuadores:** Los elementos motrices que generan el movimiento de las articulaciones pueden ser, según la energía que consuman, de tipo hidráulico, neumático o eléctrico.

- **Programabilidad:** La inclusión del controlador de tipo microelectrónica en los robots industriales permite la programación del robot de muy diversas formas. En general, los modernos sistemas de robots admiten la programación manual, mediante un módulo de programación.

3.1.2. ARDUINO

Arduino nació en 2005 en el *Ivrea Interaction Design Institute* con el fin de constituir una herramienta fácil para la creación rápida de prototipos, dirigida a estudiantes sin experiencia en electrónica y programación. Tras su rápido auge, la placa Arduino comenzó a cambiar para adaptarse a las nuevas necesidades y desafíos, diferenciando su oferta desde placas simples de 8 bits hasta productos para aplicaciones de IoT, impresión 3D y entornos integrados. Todas las placas Arduino son completamente de código abierto, lo que permite a los usuarios construirlas de forma independiente y eventualmente adaptarlas a sus necesidades particulares. El software también es de código abierto y está creciendo gracias a las contribuciones de los usuarios de todo el mundo.

¿QUE ES ARDUINO?

Arduino es una plataforma de desarrollo basada en una placa electrónica de hardware libre y fácil uso. Las placas Arduino incorporan un microcontrolador reprogramable y una serie de pines hembra a través de los cuales realizar las conexiones entre el microcontrolador y los diferentes sensores y actuadores de una manera muy sencilla (principalmente con cables Dupont). Las placas Arduino pueden leer entradas (luz en un sensor, un dedo en un botón o un mensaje de Twitter) y convertirlo en una salida, activando un motor, encendiendo un LED, publicando algo en línea. Para hacerlo, utiliza el lenguaje de programación Arduino (basado en Wiring) y el Software Arduino (IDE), basado en Processing.

Arduino está formado básicamente por una placa electrónica PCB (“PrintedCircuitBoard”, “Placa de Circuito Impreso” en español). Las PCBs son superficies planas fabricadas en un material no conductor, la cual consta de distintas capas de material conductor. Una PCB es la forma más compacta y estable de construir un circuito electrónico. Por lo tanto, la placa Arduino no es más que una PCB que implementa un determinado diseño de circuitería interna. De esta forma el usuario final no se debe preocupar por las conexiones eléctricas que necesita el microcontrolador para funcionar, y puede empezar directamente a desarrollar las diferentes aplicaciones electrónicas que necesite.

MODELOS DE PLACAS ARDUINO

Cuando hablamos de “Arduino” deberíamos especificar el modelo concreto. Se han fabricado diferentes modelos de placas Arduino oficiales [Figura 3.13], cada una pensada con un propósito diferente y características variadas (como el tamaño físico, número de pines E/S, modelo del microcontrolador, etc.). A pesar de las varias placas que existen, todas ellas pertenecen a la misma familia (microcontroladores AVR marca Atmel). Esto significa que comparten la mayoría de sus características de software, como arquitectura, librerías y documentación.



Figura 3. 13. Modelos Arduino

VENTAJAS DEL USO DE ARDUINO

Arduino es libre y extensible: así cualquiera que desee ampliar y mejorar el diseño hardware de las placas como el entorno de desarrollo, puede hacerlo sin problemas. Esto permite que exista un rico ecosistema de placas electrónicas no oficiales para distintos propósitos y de librerías de software de tercero, que pueden adaptarse mejor a nuestras necesidades.

Las ventajas más destacables del uso de Arduino son:

- **Arduino tiene una gran comunidad:** Gracias a su gran alcance hay una gran comunidad trabajando con esta plataforma. Así se genera una cantidad de documentación bastante extensa, la cual abarca casi cualquier necesidad.
- **Su entorno de programación es multiplataforma.** Se puede instalar y ejecutar en sistemas operativos Windows, Mac OS y Linux.
- **Lenguaje de programación de fácil comprensión.** Su lenguaje de programación basado en C++ es de fácil comprensión. Bajo precio.
- **Gran versatilidad.** Es un sistema reutilizable porque una vez terminado el proyecto es muy fácil poder desmontar los componentes externos a la placa y empezar con un nuevo proyecto.

3.1.3. LOS SISTEMAS DE IMPRESIÓN 3D

La impresión 3D es un grupo de tecnologías de fabricación por adición capaz de crear un objeto tridimensional mediante la superposición de capas sucesivas de un determinado material. [9] Se trata de un proceso guiado por un modelo digital para la creación de geometrías determinadas.

Los primeros equipos de impresión 3D surgieron en el año 1976 [Figura 3.14] y podían crear geometrías a través de un modelo de plástico tridimensional con un polímero fotoendurecible. Esta tecnología fue evolucionando y en 1984 surgieron varios proyectos basados en el proceso de estereolitografía, basado en añadir la adición de distintas capas mediante el curado de fotopolímeros con láseres de rayos ultravioleta. Fue en este año cuando se definió un sistema para generar objetos tridimensionales mediante la creación de un patrón del objeto a formar que dio lugar al formato de archivo STL, que es ampliamente aceptado hoy en día para la impresión 3D.



Figura 3. 14. Primer sistema de impresión 3D

En el año 1992 se desarrolló la primera máquina de impresión 3D de tipo SLA (estereolitográfico) y fue obra de la compañía 3d Systems. Un láser ultravioleta iba solidificando una fotopolímera capa por capa para crear objetos tridimensionales. Pese a no obtener piezas con un gran acabado, se puso de manifiesto la gran capacidad de esta tecnología para su evolución.

En 2006 se construye la primera impresora tipo SLS o de sintetización de láser selectivo. Una máquina que utiliza un láser para fundir materiales durante el proceso de impresión 3D y que dio una gran esperanza a la fabricación de piezas industriales, prótesis, etc.

En los últimos años la tecnología de impresión 3D se ha popularizado y ha alcanzado altas cotas de calidad que la hacen una alternativa a tener en cuenta para la creación de piezas destinadas al uso industrial.

FUNCIONAMIENTO

La impresión 3D requiere de una coordinación de softwares, hardware y materiales relacionados para la creación de elementos.

En primer lugar, el software divide el elemento en capas del mismo grosor que el diámetro de salida del material que se va a utilizar para la creación de la pieza. Para la creación de las distintas capas la impresora se va desplazando sobre los ejes x e y, mientras libera el material en las coordenadas correspondientes, de esta manera se crean geometrías en 2D. Para conseguir superponer las distintas capas y crear así objetos tridimensionales, la impresora se irá desplazando verticalmente para imprimir las distintas capas sobre las anteriores.

El primer paso que se debe realizar para poder obtener una pieza impresa mediante 3D, es obtener un archivo creado con algún software de modelado 3D.

El siguiente paso es la correcta selección del material, para lo cual actualmente los materiales más utilizados son termoplásticos como el PLA o ABS, aunque existen impresoras capaces de utilizar una amplia variedad de materiales como metales, resinas o polímeros.

Tras ello se deberá transmitir la geometría de la pieza en un formato correcto para su lectura por la impresora 3D y tras ello se podrá obtener la pieza diseñada.

METODOS DE IMPRESIÓN

Actualmente los métodos de impresión 3D más utilizados son los siguientes:

- **Impresión por inyección:** En este método, la impresora crea el modelo de capa esparciendo una capa de la sección de la pieza. Este proceso se repite hasta que se imprimen todas las capas [Figura 3.15].

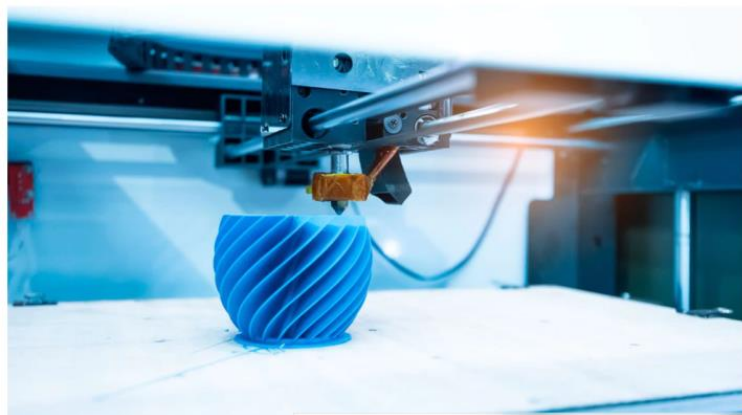


Figura 3. 15. Impresora mediante inyección

- **Modelado por deposición fundida (FDM):** Como su propio nombre indica, este método va depositando un material fundido sobre una estructura capa a capa que posteriormente es sintetizado por un láser para su solidificación. Dentro de este método se incluye también el sintetizado de metal por láser o DMLS.
- **Estereolitografía (SLA):** La tecnología estereolitografía o SLA es la que utiliza resinas líquidas fotopoliméricas que se solidifican con el uso de una luz emitida por un láser ultravioleta. De esta manera, se van creando capas de resina sólida hasta formar el objeto.

MATERIALES

Los materiales más comunes para impresión 3D son los termoplásticos que se exponen a continuación:

- **Ácido poliláctico (PLA):** Se trata de un polímero constituido por elementos similares al ácido láctico y con propiedades similares a las del tereftalato de polietileno (PET) que se utiliza habitualmente para hacer envases. Un termoplástico fabricado a partir de almidón de maíz, yuca, mandioca o caña de azúcar.
- **Acrilonitrilo butadieno estireno (ABS):** Plástico muy resistente y que aguanta altas temperaturas. Ofrece cierta flexibilidad y es fácil de pintar.
- **Filaflex:** Es un filamento elástico con una base de poliuretano y otros aditivos que ofrecen en conjunto una gran elasticidad. La impresión con este tipo de material es más lenta, pero es muy útil para determinados objetos. [Figura 3.16]

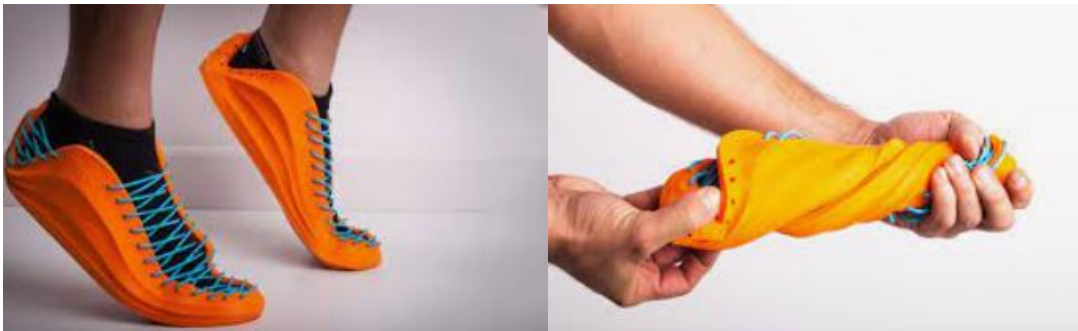


Figura 3. 16. Impresión mediante Filaflex

Existen también impresoras con capacidad para impresión de materiales de todo tipo. Algunos de ellos son el Nylon o metales amorfos (BGM), los cuales se caracterizan por tener una estructura atómica desordenada y por ello permiten adoptar varias formas para su solidificación.

APLICACIONES

Las aplicaciones que se le puede dar a esta tecnología son múltiples. Dentro de los campos que más se ha utilizado son los siguientes:

- **Medicina:** creación de prótesis [Figura 3.17], aparatos sanitarios e incluso partes del cuerpo que son tolerables por el organismo.



Figura 3. 17. Prótesis fabricada mediante impresión 3D

- **Educación:** Replica de conceptos para una mejor comprensión.
- **Industrial:** supone un gran ahorro de tiempo y costes el poder crear piezas de diferentes materiales de forma rápida.
- **Comida y alimentación:** Existen impresoras capaces de sustituir algunos procesos culinarios

En definitiva, la utilidad de la impresión 3D es amplia y hoy en día se está extendiendo incluso al uso doméstico ya que puede ser útil para la creación de complementos del hogar o juguetes.

BANCOS DE ARCHIVOS

En la actualidad hay numerosas bibliotecas online en la que se pueden obtener los diseños 3D preparados para su impresión de manera gratuita. Algunas de estas bibliotecas son GrabCad y Thingiverse [Figura 3.18].

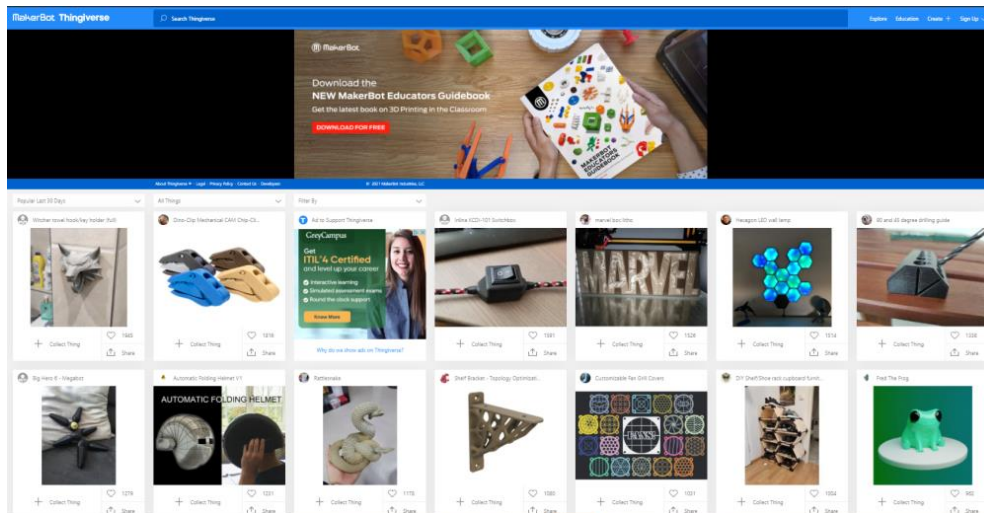


Figura 3. 18. Plataforma Thingiverse

Gracias a ello la tecnología de impresión 3D se vuelve mucho más accesible incluso para personas con nociones de diseño muy básicas.

3.1.4. LOS SERVOMOTORES

Un servomotor es un dispositivo el cual se caracteriza por su capacidad de llegar a posiciones angulares específicas de forma inmediata, siempre y cuando este dentro de su rango de operación. Para su funcionamiento el servomotor espera una serie de pulsos (señal PWM), los cuales indican el movimiento a realizar, y mientras este no presente cambio, el servo permanece en el mismo estado.

Un servomotor está compuesto por un potenciómetro, el cual va conectado al eje central del motor y tiene como función supervisar el ángulo actual en que se encuentra el motor, el circuito de control analiza el ángulo arrojado por el potenciómetro y el ángulo deseado, por lo que, de no estar en la posición correcta, el motor realiza los movimientos necesarios para poder llegar a la posición deseada.

En la siguiente imagen [Figura 3.19] se muestra un servomotor desarmado. En ella se pueden ver las partes anteriormente mencionadas.



Figura 3. 19. Servomotor desmontado

3.1.5. SOFTWARE EMPLEADO

SOLIDWORKS 2019



SOLIDWORKS es un software de diseño CAD 3D (diseño asistido por computadora) para modelar piezas y ensamblajes en 3D y planos en 2D. El software que ofrece un abanico de soluciones para cubrir los aspectos implicados en el proceso de desarrollo del producto. Sus productos ofrecen la posibilidad de crear, diseñar, simular, fabricar, publicar y gestionar los datos del proceso de diseño.

El uso de SolidWorks permite que el proceso de creación de nuevas soluciones se pueda realizar solapando sus distintas fases, con el fin de ahorrar tiempo y dinero.

SolidWorks incorpora una serie de herramientas y soluciones para las distintas fases de diseño:

- Herramientas de diseño para crear modelos y ensamblajes.
- Herramientas de diseño para la fabricación mecánica, que automatiza documentos de inspección y genera documentación sin planos 2D.
- Herramientas de simulación para evaluar el diseño y garantizar que es el mejor posible. [Figura 3.20]

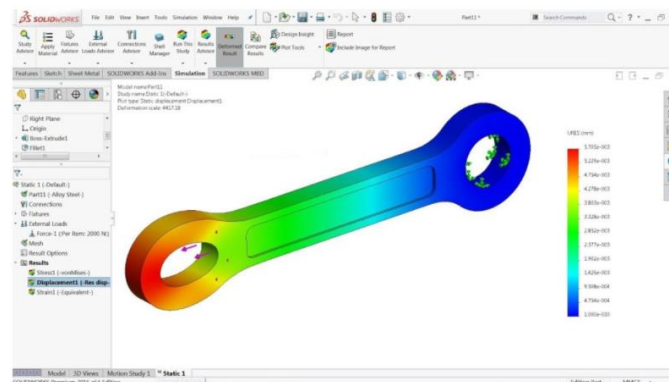


Figura 3. 20. Estudio mediante SolidWorks Simulation

- Herramientas que evalúan el impacto medioambiental del diseño durante su ciclo de vida.
- Herramientas que reutilizan los datos de CAD en 3D para simplificar el modo en que las empresas crean, conservan y utilizan contenidos para la comunicación técnica.

ULTIMAKER CURA



Se trata de un software de libre uso que, partiendo de un modelo 3D diseñado (generalmente en formato tipo STL o OBJ) es capaz de generar el código .GCODE utilizado por la impresora para realizar las trayectorias.

Cura es un software tipo slicer, cuya función es la de “trocear en capas” el modelo según las especificaciones indicadas por el usuario.

Los parámetros de impresión son definidos por el usuario dentro del software como se podrá ver más adelante en la redacción de este TFM. Estos parámetros definen tanto la estructura interna del objeto impreso, como su acabado exterior o resistencia mecánica.

ARDUINO IDE



Se necesitará de un programa específico donde poder escribir y compilar el código de control que va a ser ejecutado por el microcontrolador Arduino. Este programa será Arduino IDE (Integrated Development Environment) según sus siglas en inglés).

Se trata de un entorno muy sencillo donde se puede escribir, verificar y compilar el código. Tras ello también nos permitirá almacenarlo en la memoria del microcontrolador.

Arduino IDE proporciona también acceso a una serie de librerías que ha generado la comunidad de desarrolladores. En especial, para este proyecto, se utilizará la librería servo.h para el control de los servomotores.

ROBOTSTUDIO

RobotStudio®

El software de simulación y programación offline de ABB, RobotStudio, nos permite realizar la programación del robot a través de un PC sin necesidad de interrumpir la producción con paradas para ello.

RobotStudio se basa en *ABB Virtual Controller*. Se trata de una copia exacta del software real integrado en los robots. Esto permite realizar simulaciones muy realistas, utilizando programas de robot reales y archivos de configuración idénticos a los utilizados en el taller.

RobotStudio contiene un paquete de características y complementos que permiten una simulación offline perfecta reduciendo los riesgos, fijando las start-ups, acortando los cambios y aumentando la productividad.

3.2. ESPECIFICACIONES LOS DISTINTOS ELEMENTOS INVOLUCRADOS

En este punto se muestran las especificaciones de cada uno de los principales elementos que se van a involucrar dentro de nuestro proceso de desarrollo y fabricación de la herramienta.

Todos estos elementos estarán involucrados en el producto final de distintas maneras. Esto se podrá ver en el apartado posterior 3.4 donde se expondrá la solución seleccionada para alcanzar el diseño óptimo y también dentro del punto 4 de resultados. Es por ello por lo que en este punto no se entrará en detalle de cómo se involucra cada uno de ellos, sino que se concreta su finalidad dentro del proyecto y sus características técnicas y especificaciones como sistema individual.

1) ROBOT ABB IRB 140

El robot involucrado en este proyecto es el modelo IRB 140 de la marca ABB. Se trata de un robot de 6 ejes pequeño, rápido y potente. Se caracteriza por su versatilidad, así como por ser uno de los modelos más compactos. Soporta una carga de hasta 6 kg y tiene un alcance de 810 mm. Tiene una protección IP67 que lo hace apropiado para una extensa gama de aplicaciones.

Su diseño robusto, con cables completamente integrados, además de su completa flexibilidad y su función de detección de colisiones con completa retracción de la trayectoria, aseguran la fiabilidad y seguridad de este robot.

La versión que se encuentra dentro de las instalaciones de la Universidad de Almería es la estándar con montaje sobre plataforma elevada. [Figura 3.21]



Figura 3. 21. ABB IRB140 de la Universidad de Almería

- Especificaciones técnicas [Tabla 1]:

versión del Robot	IRB140
Alcance del 5º eje	810 mm
Capacidad de manejo	6 kg
Carga suplementaria- en la parte superior del brazo	1 kg
Carga suplementaria- en la muñeca	0,5 kg
Número de ejes	6
Protección	IP67
Montaje	Cualquier ángulo
Controlador	IRC5 Compact
Suministro de señal integrado	12 señales en el brazo superior

Tabla 1. Especificaciones técnicas IRB140

- Rendimiento (según ISO 9283) [Figura 3.22]:

Versiónes de robots	IRB 140
5 kg lado de recogida	
25 x 300 x 25 mm	0.85 s
Max. TCP velocidad	2.5 m/s
Max. TCP aceleración	20 m/s ²
Tiempo de aceleración 0-1 m/s	0.15 s
Posición de repetibilidad	0.03 mm

Figura 3. 22. Rendimiento ABB IRB 140

- Dimensiones [Figura 3.23]:

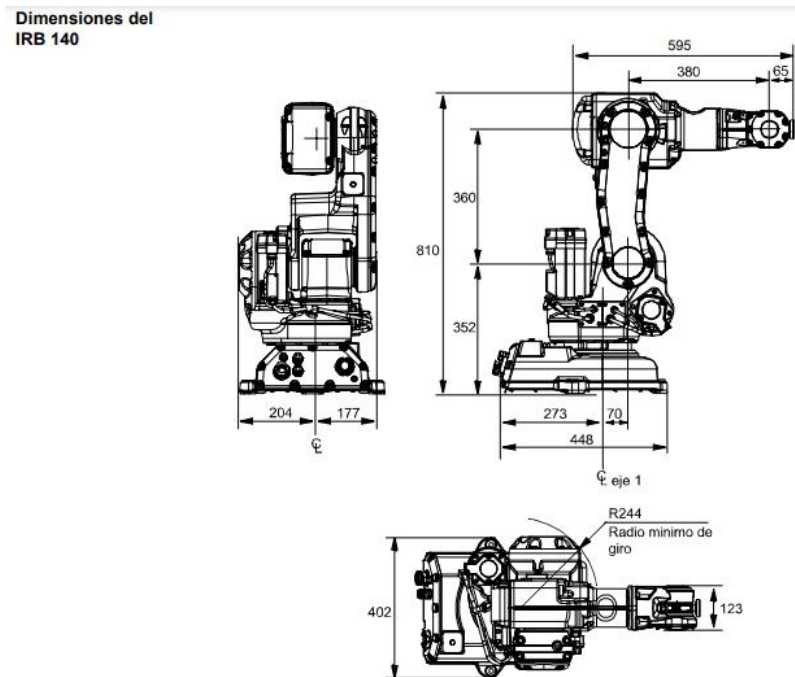


Figura 3 Vistas posterior, lateral y superior del manipulador (dimensiones en mm).

Figura 3. 23. Dimensiones ABB IRB 140

- Área de trabajo [Figura 3.24]:

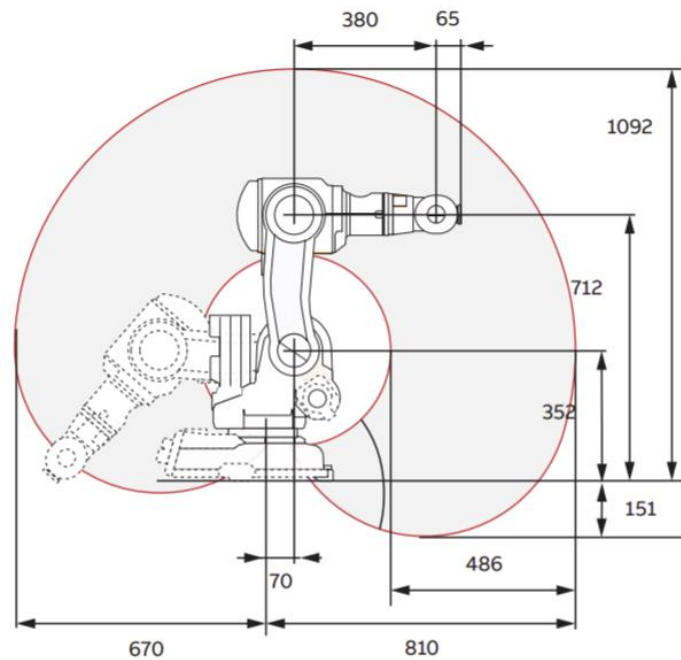


Figura 3. 24. Área de trabajo IRB140

- Información técnica [Figura 3.25]:

Electrical Connections			
Supply voltage	200-600 V, 50/60 Hz		
Rated power transformer rating	4.5 kVA		
Power consumption	0.4 kW		
Physical			
Robot base	400 x 450 mm		
Robot controller	950 x 800 x 620 mm		
Robot manipulator weight	98 kg		
Environment			
Ambient temperature for robot manipulator			
During operation	+5° C (41° F) to + 45° C (113° F)		
Relative humidity	Max. 95%		
Options	Foundry Plus 2 SteamWash (High pressure steam washable) Clean Room, class 6 (certified by IPA)		
Noise level	Max. 70 dB (A)		
Safety	Double circuits with supervisions, emergency stops and safety functions. 3-position enable device		
Emission	EMC/EMI shielded		
Movement			
Axis movement	Working range	Velocity* IRB 140	Velocity* IRB 140T
Axis 1 rotation	+360°	200°/s	250°/s
Axis 2 arm	+200°	200°/s	250°/s
Axis 3 arm	+280°	260°/s	260°/s
Axis 4 wrist	Unlimited (Default: 400°)	360°/s	360°/s
Axis 5 bend	+230°	360°/s	360°/s
Axis 6 turn	Unlimited (Default: 800°)	450°/s	450°/s

Figura 3. 25. Ficha técnica IRB140

- Características del terminal de acople a la herramienta [Figura 3.26]:

Estos datos son de especial interés ya que nuestra herramienta ha de acoplarse correctamente a este terminal.

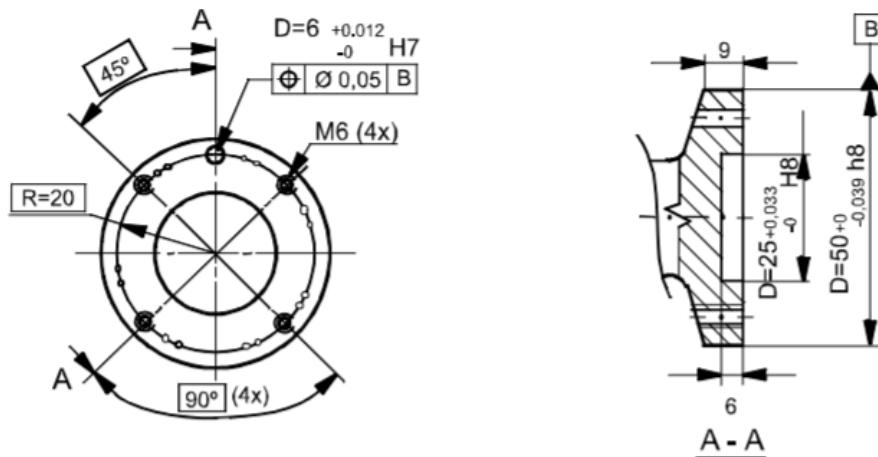


Figura 3. 26. Terminal de acoplamiento muñeca IRB 140

2) CONTROLADOR IRC5C DEL ROBOT [Figura 3.27]



Figura 3. 27. Controlador ABB IRC5C

Nuestro robot estará controlado a través del robótico industrial IRC5C compacto de la misma marca ABB. Este controlador nos posibilita la maniobrabilidad del robot y a través de él se implementará el proceso de manipulación de objetos final.

El IRC5C ofrece el control de movimiento del robot ABB, así como gran flexibilidad, seguridad, modularidad, interfaces de aplicación, control de múltiples robots y compatibilidad con herramientas de PC.

Entre las muchas características de este controlador se podrían destacar las siguientes:

- Rápido y preciso
- Seguro
- Compatible
- Programable
- De confianza

A demás de ello, el controlador IRC5C:

- Ofrece beneficios de ahorro de espacio y una puesta en marcha sencilla a través de una entrada de alimentación monofásica.
- Conectores externos para todas las señales y un sistema de E/S expansible integrado de 16 entradas y 16 salidas.
- Flexpendant: sistema basado en una pantalla táctil portátil y un joystick para conseguir una interacción intuitiva con el robot.

Información técnica IRC5C [Figura 3.28]

Conexiones Eléctricas	
Tensión de alimentación	Fase simple 220/230 V, 50-60 Hz
Físico	
Dimensiones	320 x 449 x 442 mm
Peso	28,5 kg
Ambiente	
Temperatura ambiente	0-45°C
Humedad relativa	Max. 95% sin condensación
Seguridad	Sistema de seguridad doble canal Certificado según ISO10218
Seguridad extendida	SafeMove2 <ul style="list-style-type: none"> ▪ Herramienta de supervisión de posición, velocidad y orientación ▪ Campo de bus ▪ Supervisión de la posición y la velocidad del eje ▪ Supervisión en reposo
Protección	IP20

Figura 3. 28. Información técnica IRC5C

Información técnica Flexpendant [Figura 3.29]

Ambiente	
Funciones	Pantalla táctil gráfica a color Palanca de mando Conexión en caliente: agregar / quitar durante la operación Teclado de membrana con 12 botones Soporte de memoria USB
Funciones de seguridad	Parada de emergencia Interruptor de habilitación de 3 posiciones (circuito doble)
Protección	IP54

Figura 3. 29. Información técnica Flexpendant

3) MÓDULO DSQC 653 DE ENTRADAS/SALIDAS

Se trata del módulo de entradas y salidas que nos permitirá la programación conjunta de nuestro sistema.

Se encuentra acoplado junto con el controlador IRC5C del robot y nos proporcionará las salidas necesarias para programar correctamente todo el conjunto.

- Datos técnicos [Tabla 2]:

Digital inputs	<p>Number of digital inputs: 8</p> <p>Rated voltage:</p> <ul style="list-style-type: none"> • 24 VDC <p>Input voltage range:</p> <ul style="list-style-type: none"> • "1" ---> 15 to 35 V • "0" ---> -35 to 5 V <p>Input current at rated voltage:</p> <ul style="list-style-type: none"> • Typical ---> 5 mA (approx) <p>Switch-over level:</p> <ul style="list-style-type: none"> • Typical --->12V (approx) <p>Delay (with default filtering):</p> <ul style="list-style-type: none"> • Typical --->5 ms • minimum --->4 ms • maximum --->6ms <p>Power dissipation or channel at rated voltage:</p> <ul style="list-style-type: none"> • Typical --->150 mW (approx)
Relay outputs	<p>Number of relay outputs: 8</p> <p>Single pole with one make contact (normally open)</p> <p>Rated voltage</p> <ul style="list-style-type: none"> • 24 VDC/120 VAC rms <p>Rated output current</p> <ul style="list-style-type: none"> • 2 A/channel <p>Output load</p> <ul style="list-style-type: none"> • minimum 2.5 VA/channel <p>Delay time on board</p> <ul style="list-style-type: none"> • Typical - 6 ms (make contact), 7 ms (break contact)
Supply voltage	24 VDC
Supply source	24 V I/O or separate external supply
External supply for relay outputs	<p>Voltage</p> <ul style="list-style-type: none"> • 19 - 35 VDC, 24 - 140 VAC rms
SW connections	<p>Support for the following connections:</p> <ul style="list-style-type: none"> • Polled • Change-Of-State • Change-Of-State with acknowledge suppression • Cyclic • Cyclic with acknowledge suppression <p>For descriptions of the different types of I/O connections, see I/O messages - connection types on page 15.</p>

Tabla 2. Datos técnicos módulo DSQC 653

4) IMPRESORA 3DBQWITBOX 2 [Figura 3.30]



Figura 3. 30. Impresora 3D BQ WitBox 2

Un elemento clave para poder fabricar la herramienta objeto de este TFM será la impresora 3D a través de la cual se obtendrán todas las piezas diseñadas.

El modelo de impresora 3D con el cual se va a trabajar es el WitBox 2 de la marca BQ. Esta impresora se caracteriza por su capacidad para trabajar en entornos domésticos, profesionales y educativos con resultados extraordinarios. Se trata de una impresora compacta con un compartimento cerrado que permite imprimir figuras con gran calidad ya que está aislada de los cambios de temperatura exteriores, así como de las corrientes de aire. Cuenta con un extrusor especialmente diseñado para evitar las obstrucciones. Al reducirse el riesgo de obstrucciones, puede imprimir en una mayor variedad de filamentos, como PLA, bronce, madera o cobre, e, incluso, materiales flexibles como el Filaflex.

Esta impresora cuenta además con un sistema de auto nivelado que permitirá una impresión más sencilla y rápida.

Especificaciones [Tabla 3]:

Velocidad de impresión Máxima	200 mm/s
Temperaturas de funcionamiento óptimo	15 °C - 25 °C.
Resolución	20 micras
Peso	34kg
Cama caliente	NO
Dimensiones Impresora con bobina PLA	(x)508,06 x (y)485,08 x (z)461 mm
Área de impresión	(x)297 x (y)210 x (z)200 mm
Grosor Filamentos	1,75 mm de diámetro
Materiales Filamentos	PLA, madera, bronce, cobre y Filaflex.
Mecánica extrusor	<ul style="list-style-type: none"> - Extrusor de diseño propio con Double Drive Gear, ajuste de tracción y guiado de filamento para máxima compatibilidad con filamentos flexibles/blandos -Hot-End diseñado para un fácil mantenimiento -Boquilla de 0,4 mm para filamento de 1,75 mm -Disipador de aletas con ventilador axial controlado por Firmware -Tobera de refrigeración impresa
Electrónica	<ul style="list-style-type: none"> -Controladora BQ Zum CNC -BQ Zum Inductive Sensor para asistencia en la nivelación de la cama -BQ Zum LCD Smart Controller -Fuente de alimentación 12V 150W -Termistor 100k en el Hot-End -Cartucho calefactor 12V 40W -Cableado optimizado mediante fajas
Software	Autonivelación, nivelación manual mediante sensor inductivo diseñado exclusivamente para impresión 3D, modo de inactividad
Entorno recomendado	Cura Software, Slic3r.
Archivos admitidos	Gcode
Sistemas Operativos compatibles	Windows XP y superiores, Linux y Mac OS X y superiores
Comunicaciones	Lector de tarjetas SD estándar Puerto USB tipo B

Tabla 3. Especificaciones BQ Witbox2

5) SERVOMOTOR ESTÁNDAR HD-3001HB [MMSV003]

Se trata de un servomotor de pequeñas dimensiones con una rotación de 180° pensado para el accionamiento de pequeños sistemas [Figura 3.31]. Este motor eléctrico está pensado especialmente para poder ser controlado a través de Arduino mediante bibliotecas específicas como servo.h.

Dentro de nuestro diseño se utilizará este servomotor para proporcionar a la pinza el movimiento de apertura y cierre de los brazos que forman la pinza.



Figura 3. 31.Servomotor HD-3001HB

Especificaciones [Tabla 4]:

Tipo de Modulación	analógica
Dimensiones	40.7 x 20.5 x 39.5 mm
peso	43 g± 1g
Par máximo	@ 4.8V: 3.5 kg·cm @ 6V: 4.4 kg·cm
Velocidad	@ 4.8V: 0.15 sec/60° @ 6V: 0.12 sec/60°
Par de bloqueo (en bloqueado)	@ 4.8V: 3.5 kg·cm @ 6V: 4.4 kg·cm
Corriente de bloqueo (bloqueado)	@ 4.8V: 800 mA @ 6V: 900 mA
Corriente inactiva (en parada)	@ 4.8V: 4 mA @ 6V: 5 mA
Ángulo límite	180° ± 10°
Radio de reducción	1/240
Dirección de rotación	Counterclockwise (when 1500P2000 μsec)
Rango de operación (Voltaje)	4,8 – 6V

Tabla 4. Especificaciones servomotor HD-3001HB

6) SISTEMA DE VENTOSA CON BOMBA DE VACÍO PARA SUCCIÓN [FIGURA 3.32]



Figura 3. 32. Sistema de ventosas para succión

Como ya se ha explicado anteriormente en los requisitos de diseño, la herramienta ha de ser capaz de manipular objetos con geometrías planas que no puedan ser manipuladas mediante el cierre de la pinza. Es por ello por lo que para el diseño de la herramienta se contará con un sistema de ventosas accionado a través de una bomba de vacío y una válvula electrónica de 3 vías.

Las especificaciones del Kit de bomba de vacío con ventosa disponible son las siguientes:

Parámetros del conjunto [Tabla 5]:

Parámetros de la bomba de aire:	Parámetros válvula electrónica:	Parámetros del producto general
<ul style="list-style-type: none"> -Voltaje nominal: CC 4,5 V -Corriente sin carga: 0.35A -Voltaje Compatible: CC 3V-5V -Rango de presión: 400-650 mmhg -Grado máximo de vacío: -350 mmhg -Peso: 60g 	<ul style="list-style-type: none"> -Voltaje nominal: CC 6V -220mA -Voltaje Compatible: cc 5V-6V -Potencia: <2W -Rango de presión: 0-350 mmhg -Fluido utilizable: aire -Longitud del Cable: 15cm -Peso: 16g 	<ul style="list-style-type: none"> -Rango de voltaje: 3,5-7,4 V -Corriente máxima: 2A -Corriente de funcionamiento: 1A -Protección contra el exceso de temperatura: corta automáticamente la fuente de alimentación y deja de funcionar

Tabla 5. Parámetros sistema ventosas

7) SENSOR DE DISTANCIA VL53L0X

Se incluirá dentro del diseño un sensor de distancia para indicar la proximidad a objetos y de este modo obtener una mayor información disponible de la herramienta.

El sensor es el VL53L0X y se trata de un sensor compacto cuyo funcionamiento se basa en la medición del tiempo de vuelo del láser incorporado (ToF: Time of Flight). Proporciona una medición de distancia precisa sea cual sea la reflectancia del objetivo. Puede medir distancias absolutas hasta 2 m. El principio de funcionamiento de los sensores de distancia por tiempo de vuelo es similar al de los sensores ultrasonido: El sensor posee un emisor laser que cada cierto tiempo emite un haz de luz, al encontrarse con un objeto la luz es reflejada, el sensor se encarga de medir el tiempo entre la emisión y la detección de la luz y como la velocidad de la luz es un valor conocido entonces se puede calcular la distancia al objeto con el tiempo de vuelo.

El emisor VCSEL de 940 nm del VL53L0X es totalmente invisible para el ojo humano y permite un alcance más largo y mayor inmunidad a la luz ambiental.

Especificaciones técnicas [tabla 6]:

PARÁMETROS SENSOR DE DISTANCIA VL53L0X	
-Voltaje de alimentación: 3V - 5V DC	-Puede operar en ambientes con niveles altos de luz infrarroja
-Consumo corriente: 10mA (40mA máx)	-Laser clase 1 (Estándar IEC 60825-1:2014)
-Rango de medición: de 50 mm a 1200 mm	-No causa daños a la vista humana
-Precisión: +-30 mm	-Óptica anti-interferencias embebida
-Interfaz digital: I2C 400kHz	-No necesita de óptica adicional para funcionar
-I2C compatible con sistemas de 3.3V y 5V	-Solo necesita una fuente de voltaje
-Salida dato I2C de 16 bits con la distancia en milímetros	-Xshutdown (reset) y pin de interrupción
-Incluye regulador de voltaje en placa	-Dirección I2C programable (por defecto: 0x52)
-Laser VCSEL de 940nm	-Dimensiones del sensor: 4.4*2.4*1.0 mm
-Driver VCSEL	-Dimensiones del módulo: 10.5*13.3 mm
-Sensor de medición con microcontrolador embebido	
-Driver VCSEL	
-Sensor de medición con microcontrolador embebido	

Tabla 6. Especificaciones técnicas sensor de distancia

8) MICROPROCESADOR (ARDUINO)

Para el desarrollo del proyecto se ponen a disposición 2 modelos distintos de esta herramienta. Cabe destacar que los modelos que se presentarán dentro de esta memoria como Arduino, no pertenecen a esta marca comercial, sino que se han utilizado sus versiones correspondientes de otras marcas comerciales más económicas. Sin embargo, puesto que la funcionalidad idéntica, a efectos de realizar un desarrollo más claro del proceso, se presentan en este punto las versiones de la marca Arduino correspondientes a sus equivalentes utilizados.

A través de los distintos Arduino que se incluirán dentro del sistema diseñado se podrá realizar el control y automatización de las operaciones de la herramienta.

Arduino Uno:



Figura 3. 33. Arduino Uno

La placa Arduino Uno [Figura 3.33] se trata de una placa electrónica basada en el chip de Atmel ATmega328. Tiene 14 pines digitales de entrada / salida, es el Arduino Pinout de los cuales 6 los puede utilizar como salidas PWM, 6 entradas analógicas, un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, una cabecera ICSP y un botón de reset.

Especificaciones técnicas [tabla 7]:

Microcontrolador	ATmega328
Voltaje de operación	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (límites)	6-20V
Pines de E/S digitales	14 (de los cuales 6 proporcionan salida PWM)
Pines de entrada analógica	6
Corriente DC por pin de E/S	40 mA
Corriente DC para 3.3V Pin	50 mA
Memoria Flash	32 KB de los cuales 0,5 KB utilizados por el bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Velocidad de reloj	16 MHz

Tabla 7. Especificaciones Arduino Uno

PINOUT DE ARDUINO UNO

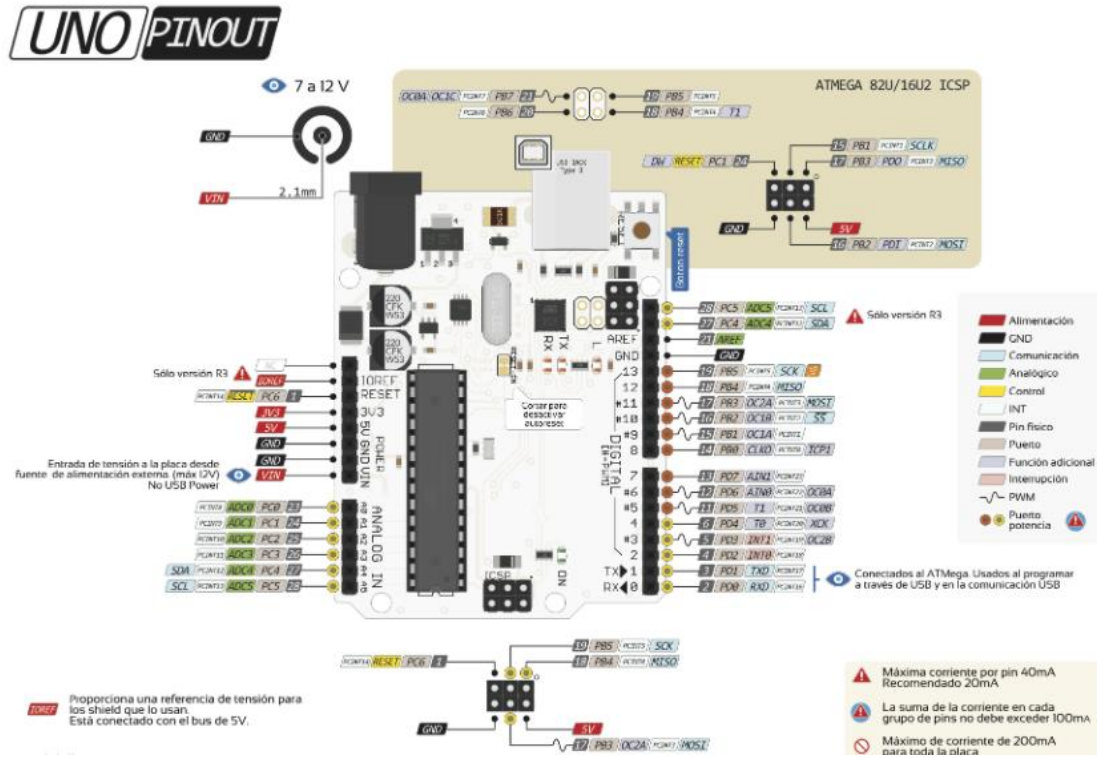


Figura 3. 34. Disposición de pines Arduino Uno

Arduino Nano:

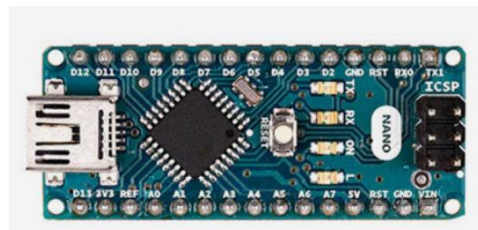


Figura 3. 35. Arduino Nano

Arduino Nano [Figura 3.35] es una placa microcontroladora pequeña, compatible, flexible y fácil de usar, desarrollada por Arduino.cc en Italia, basada en ATmega328p (Arduino Nano V3.x) / Atmega168 (Arduino Nano V3.x). Sus funcionalidades son las mismas que en Arduino UNO, pero en tamaño más pequeño. Su tensión de funcionamiento es de 5V y la de entrada puede variar de 7 a 12V. Su Pinout contiene 14 pines digitales, 8 pines analógicos, 2 pines de reinicio y 6 pines de potencia. Cada uno de estos pines digitales y analógicos tiene asignadas múltiples funciones, pero su función principal debe configurarse como entrada o salida.

La placa Arduino Nano no cuenta con una toma de alimentación DC como otras placas Arduino, sino que tiene un puerto mini-USB. Este puerto se utiliza tanto para la programación como para la monitorización en serie.

Especificaciones técnicas [Tabla 8]:

Microcontrolador	ATmega328
Voltaje de operación	5V
Voltaje de entrada (recomendado)	7-12V
Arquitectura	AVR
Pines de E/S digitales	22
Pines de entrada analógica	8
Salida PWM	6
Corriente DC por pin de E/S	40 mA
Memoria Flash	32 KB de los cuales 2 KB utilizados por el bootloader
SRAM	2 KB
EEPROM	1 KB
Velocidad de reloj	16 MHz
Consumo de energía	19 mA
Tamaño de la placa de circuito impreso	18 x 45 mm
Peso	7 g

Tabla 8. Especificaciones Arduino Nano

PINOUT DE ARDUINO NANO

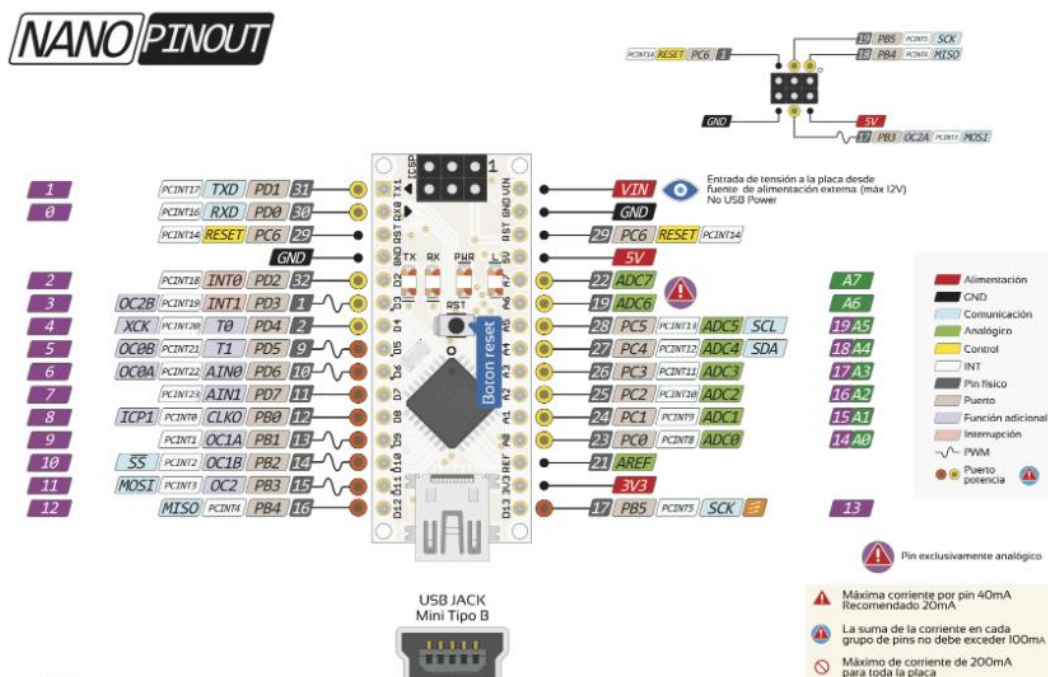


Figura 3. 36. Disposición de pines Arduino Nano

9) MÓDULO ARDUINO SENSOR SHIELD V5.0

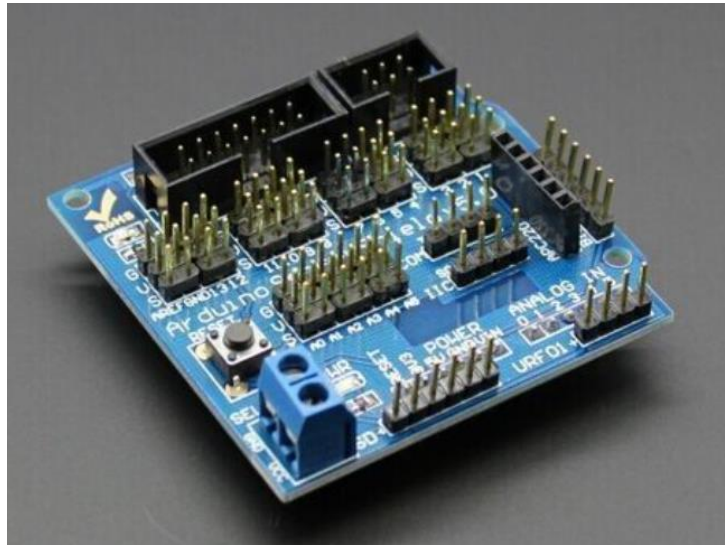


Figura 3. 37. Arduino sensor Shield V5.0

Esta placa de expansión para Arduino UNO es un SensorShield [Figura 3.37], esto es, pensada para conectar fácilmente multitud de módulos de expansión (sensores, actuadores, motores, etc.) a nuestro controlador. Esta placa es de especial interés dentro de este proyecto ya que se necesitarán varias entradas para conectar los distintos componentes que harán funcionar la herramienta (servomotor para cierre mecánico de las pinzas, servomotor y válvula para el uso de las ventosas).

La principal ventaja de este tipo de placas es que todas las señales se presentan en un grupo de 3 pines:

- G (Ground): Masa
- V (Voltage): Alimentación
- S (Signal): Señal

Si queremos conectar un módulo, tendremos todos los pines de Entrada/Salida accesibles y siempre disponibles pines de alimentación donde pincharemos nuestros cables.

Esto suponía un problema con el uso de la placa de Arduino UNO directamente, pues con 3 sensores el número de conexiones de masa y alimentación es insuficiente y se necesitaría usar una placa de prototipado.

Especificaciones [Tabla 9]:

Voltaje de operación	5V
Compatible	Arduino UNO, MEGA, LEONARDO, DUE
puertos I/O	14
Puertos de Entrada Analógica	22
Puertos	<ul style="list-style-type: none"> • UART • I2C • módulo inalámbrico APC220 • módulo Bluetooth HC06 o HC05 • módulo SD card • URF01+ • LCD paralelo • LCD serial
Led	Power y Led L (Pin 13)
Pulsador	Reset

Tabla 9. Especificaciones Sensor Shield V5.0

Pinout de Sensor Shield V5.0 [Figura 3.38]:

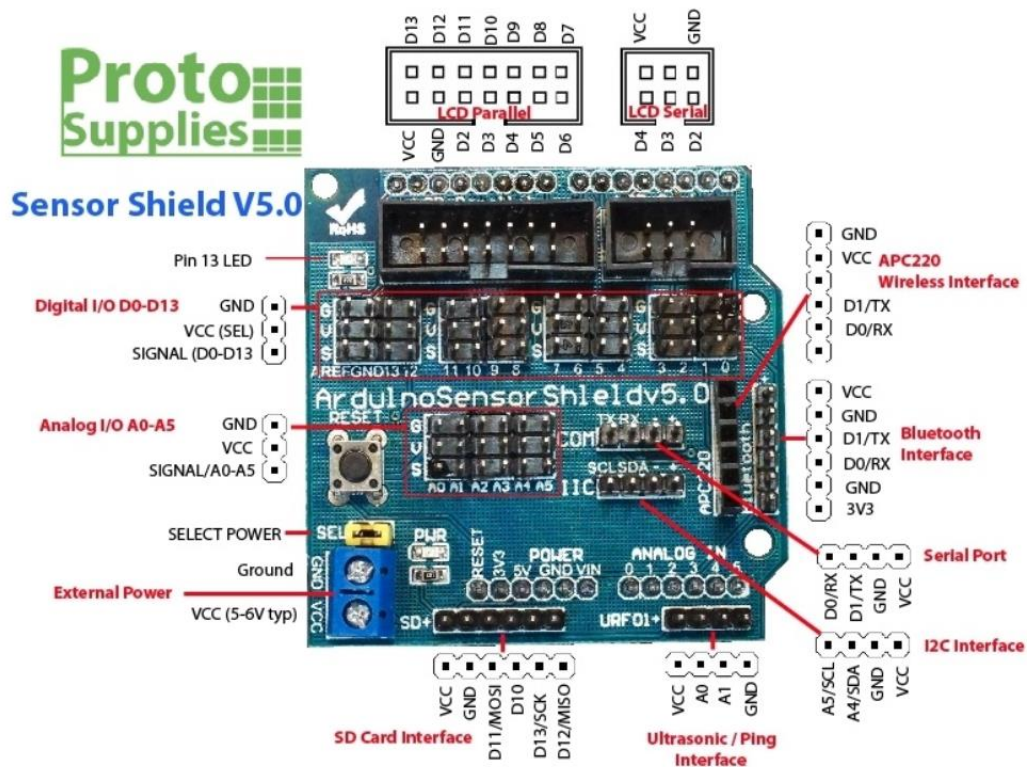


Figura 3. 38. Disposición de pines Sensor Shield v5.0

10) MÓDULOS DE COMUNICACIÓN ARDUINONRF24L01 [Figura 3.39]

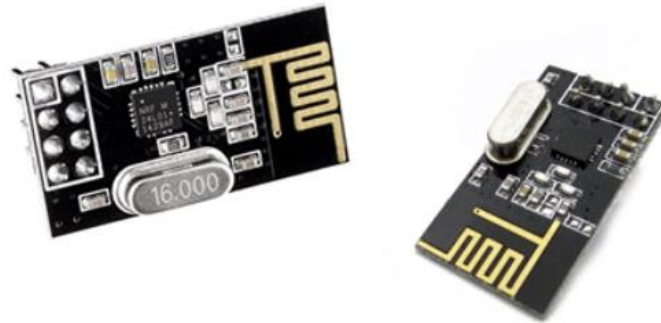


Figura 3. 39. Módulo de comunicación NRF24L01

Con el fin de dotar al sistema de la capacidad de recibir y enviar instrucciones entre los distintos componentes encargados de la automatización del proceso, se ha incluido el módulo de radiofrecuencia para poder comunicarlos entre sí. De esta manera, se podrán enviar instrucciones de un Arduino a otro sin necesidad de conexiones físicas que interfieran en el diseño. En el apartado 3.4 se explicará con mayor detalle el uso correspondiente a este módulo de comunicación.

El NRF24L01 es un chip de comunicación inalámbrica que podemos conectar a un procesador Arduino. Permite el establecimiento de conexiones inalámbricas entre placas Arduino a una distancia de entre 10 y 25 metros, dependiendo de los obstáculos interpuestos.

El NRF24L01 integra un transceptor RF (transmisor + receptor) a una frecuencia entre 2.4GHz a 2.5GHz, una banda libre para uso gratuito. La velocidad de transmisión es configurable entre 250 Kbps, 1Mbps, y 2 Mbps y permite la conexión simultánea con hasta 6 dispositivos.

PinoutNRF24L01 [Figura 3.40]

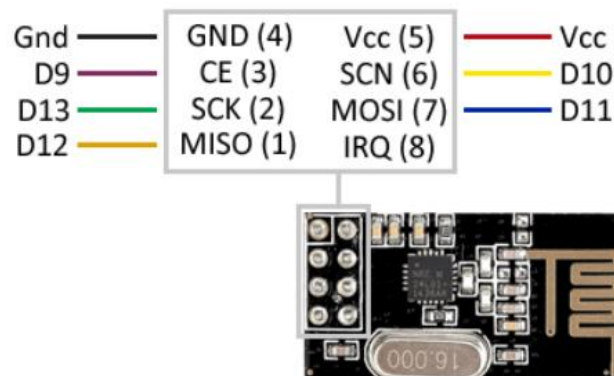


Figura 3. 40. Pinout nRF24L01

11) PANTALLA LCD Y MÓDULO DE COMUNICACIÓN I2C [Figura 3.41]

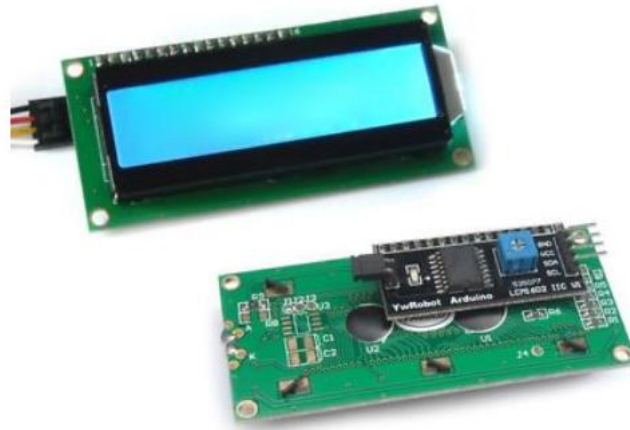


Figura 3. 41. Pantalla LCD y módulo de comunicación I2C

Dentro de nuestro diseño se incluirá una pantalla LCD que nos permitirá ver los datos proporcionados por el sensor de distancia e irá incluida de manera compacta dentro de la herramienta pinza.

Se utilizará una pantalla LCD Alfanumérica modelo 1602A cuyas características técnicas son:

- Modo de pantalla: STN, BLUB
- Formato de la pantalla: 16 Character x 2 Line
- Input Data: 4-Bits or 8-Bits interfaceavailable
- Voltaje de alimentación: 5V±10%
- BACKLIGHT: LED

Junto con la pantalla se incorporará el módulo de comunicación I2C para conseguir la correspondiente comunicación con Arduino.

Esquema de conexión del módulo I2C [Figura 3.42]:



Figura 3. 42. Conexiones del módulo I2C para pantalla LCD

3.3. ESTUDIO DE LAS POSIBLES SOLUCIONES

Al inicio del desarrollo de este proyecto se han estudiado varias soluciones para resolver cada uno de los problemas de diseño, fabricación y puesta en marcha de los distintos componentes. En este apartado se expondrán las distintas soluciones estudiadas para cada uno de ellos y sus puntos a favor y en contra para adoptarlos en el diseño final.

No se entrará en detalle de las geometrías finales de cada componente o sus interconexiones, si no que este punto muestra el proceso de estudio de las distintas soluciones tipo. Posteriormente se desarrollará la solución escogida, la cual culminará en el diseño final de la herramienta tras su correspondiente fase de desarrollo.

3.3.1. GEOMETRÍA DEL MECANISMO TIPO PINZA.

El primer problema que se ha encontrado para el diseño de la herramienta es el de conseguir un sistema de agarre de elementos mediante pinza que pueda ser accionado con el movimiento del servomotor HD-3001HB. Se deberá tener en cuenta también que dentro de esta misma herramienta ha de ir el sistema de agarre mediante ventosas y no podrá haber interferencias entre ellos.

Las soluciones estudiadas son las siguientes:

- Sistema de accionamiento de pinza mediante cremallera [Figura 3.43]:
 - o Ventajas: diseño sencillo y fácil transmisión de potencia desde el servomotor a la herramienta.
 - o Desventajas: en este diseño es necesario que una de las dos secciones de la pinza se encuentre fija en el frontal de la herramienta, lo cual interferirá con la colocación del sistema de agarre mediante ventosas.

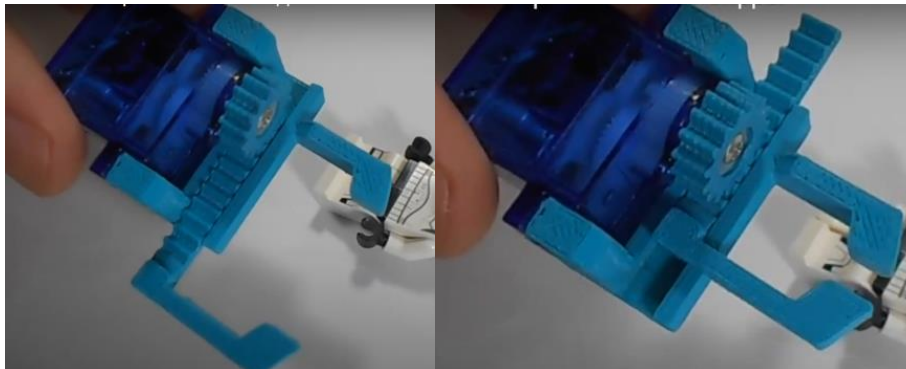


Figura 3. 43.Sistema pinza mediante cremallera

- Sistema de doble cremallera para accionamiento de las pinzas [Figura 3.44]

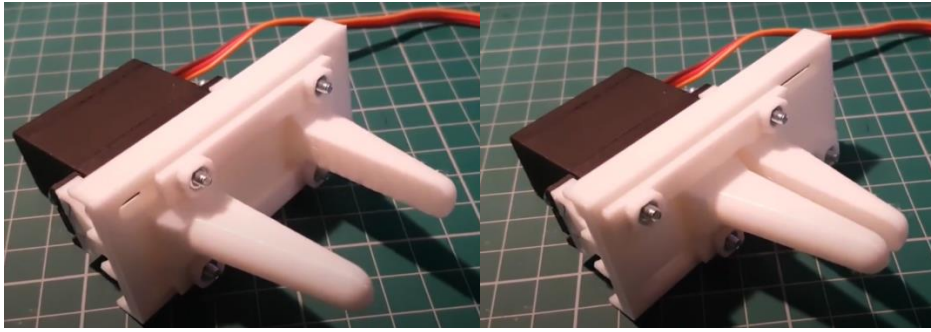


Figura 3. 44. Sistema pinza de doble cremallera

- o Ventajas: Al igual que el sistema anterior tiene la ventaja de un diseño sencillo y fácil transmisión de potencia desde el servomotor a la herramienta.
 - o Desventajas: Presenta el problema de que en la posición de cierre de las pinzas, el espacio en el plano intermedio para la colocación del sistema de ventosas desaparece.
- Mecanismo tipo pinza simple con transmisión directa [Figura 3.45].



Figura 3. 45. Sistema pinza de transmisión directa

- o Ventajas: Facilidad de acople para la transmisión del movimiento de giro del servomotor a los brazos de la pinza. Geometría válida para no interferencia entre el sistema de pinza y el mecanismo de ventosa.
- o Desventajas: su principal desventaja es que debido a la geometría de la pinza, la cara que entrará en contacto con la pieza a manipular no siempre va a encontrarse paralela al plano medio de la pieza, dificultando así su manipulación. Además existe la posibilidad de giro excesivo por parte de las pinzas debido al rango de movimiento del servomotor (180°). En el caso de error en la programación el sistema podría abrir en exceso las pinzas e impactar contra los demás elementos de control que han de ser incluidos en la herramienta.

- Mecanismo de 4 barras con sistema de transmisión directa del servomotor a sistema de pinzas [Figura 3.46]:

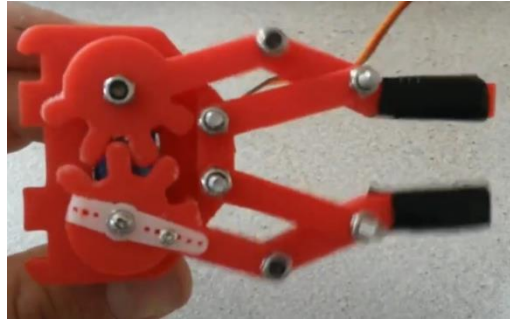


Figura 3. 46.Sistema pinza de transmisión directa con mecanismo de 4 barras

- Ventajas: Facilidad de acople para la transmisión del movimiento de giro del servomotor a los brazos de la pinza. La geometría del sistema de 4 barras permite diseñar el mecanismo para que no interfiera con el sistema de ventosas. Su geometría también permite el diseño de la pinza para que la parte plana que entrará en contacto con la pieza a manipular, siempre se encuentre paralela al plano medio de la herramienta, facilitando la manipulación de la misma.
- Desventajas: posibilidad de giro excesivo por parte de las pinzas debido al rango de movimiento del servomotor (180°). En el caso de error en la programación el sistema podría abrir en exceso las pinzas e impactar contra los demás elementos de control que han de ser incluidos en la herramienta.

- Mecanismo de 4 barras con transmisión indirecta. [Figura 3.47]:

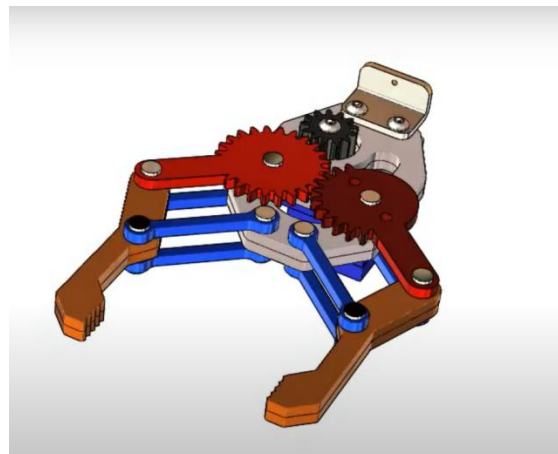


Figura 3. 47. Sistema pinza con transmisión indirecta mediante mecanismo de 4 barras

- Ventajas: La geometría del sistema de 4 barras permite diseñar el mecanismo para que no interfiera con el sistema de ventosas. Se elimina la posibilidad de colisión por error de programación, ya que se podrá incluir una relación de reducción entre el engranaje que

Diseño y fabricación de una herramienta multifunción para un robot industrial.

transmite la potencia del servomotor y el engranaje conducido del sistema de 4 barras. Su geometría también permite el diseño de la pinza para que la parte plana que entrará en contacto con la pieza a manipular siempre se encuentre paralela al plano medio de la herramienta, facilitando la manipulación de la misma.

- Desventajas: Diseño de mayor complejidad.

3.3.2. SISTEMA DE CONTROL

Para llevar a cabo el funcionamiento del sistema será necesario implementar una fase de control capaz de enviar las instrucciones y ejecutar cada una de las acciones necesarias.

Para implementar esta fase de control nuestro diseño contará con una serie de sistemas programados e interconectados entre sí, capaz de realizar todas las acciones necesarias para cumplir los requisitos iniciales.

Control del robot

Para el control del robot ABB tan solo disponemos de la posibilidad de llevarlo a cabo mediante su controlador IRC5C que se encuentra junto a él.

Se deberán programar mediante lenguaje Rapid, a través de RobotStudio, todas las funcionalidades del robot. Esto incluirá sus movimientos y el control de las salidas del módulo E/S DEQC 653 para hacer funcionar la herramienta pinza-ventosa.

Control de la herramienta acoplada al robot

Para el control de la herramienta se han estudiado 3 posibilidades, en todas ellas existe el factor común de que será un conjunto de placas Arduino el encargado de controlar a los actuadores finales que manejarán las acciones de la herramienta acoplada al robot. Es por ello por lo que la configuración y número de estas placas Arduino se explicará en mayor detalle dentro del punto de la solución seleccionada.

Las 3 distintas opciones que se han estudiado son:

- **Sistema controlado por el operario mediante Panel manual + conjunto de placas Arduino:**

Mediante este método se realizará un medio de control en el cual sea el operario el que seleccione las acciones que va a ejecutar la herramienta. De esta manera el control de la misma queda únicamente bajo la supervisión del usuario. Este método de control será una placa controlada por Arduino y que active y envíe las distintas señales de control mediante el accionamiento de interruptores manuales.

Diseño y fabricación de una herramienta multifunción para un robot industrial.

Una vez el operario seleccione las acciones, estas se llevarán a cabo mediante el conjunto de placas Arduino programadas para transmitir la información y ejecutar la acción correspondiente. Estas placas Arduino irán conectadas con los actuadores correspondientes para así poder ejecutar las acciones que se les indique.

- Sistema controlado por el controlador del robot + conjunto de placas Arduino:

En este caso en lugar de un operario, será el propio controlador IRC5C del robot el cual, a través de su programación, seleccionará la acción que se ha de llevar a cabo con la herramienta. Este sistema tiene la ventaja de ser totalmente autónomo y no necesitar de la intervención de un operario, sin embargo, esto restará flexibilidad al sistema puesto que el único ciclo de trabajo posible será el implementado en la programación del robot.

Al igual que en el sistema anterior, la transmisión y ejecución de las acciones será llevado a cabo por un conjunto de sistemas Arduino interconectados entre sí y a los actuadores correspondientes.

- Sistema mixto con posibilidad de control por operario y robot + conjunto de placas Arduino

Esta solución es una combinación de las 2 anteriores, en la cual se implementará el programa completo de control dentro del controlador del robot, pero también se permitirá la actuación puntual de un operador.

Esto se conseguirá mediante una placa controlada por Arduino en la que, a través de la activación de las salidas del controlador del robot, el Arduino detecta esa información y transmite las instrucciones necesarias a los actuadores para llevar a cabo las diferentes acciones. Se incorporará además la opción de activar manualmente las salidas correspondientes a las enviadas por el controlador del robot, de manera que también se podrá manejar manualmente el sistema mediante interruptores.

Esta opción de control aporta una mayor flexibilidad, ya que el funcionamiento de la herramienta estará completamente automatizado, pero también dejará la posibilidad de que un operario realice acciones con la herramienta en momentos puntuales.

3.4. SOLUCION SELECCIONADA

Geometría del mecanismo tipo pinza.

Tras el estudio de las distintas opciones se ha seleccionado la solución del mecanismo de 4 barras con transmisión indirecta. [Figura 3.47]

Esta opción ha sido la seleccionada ya que, aunque su diseño sea un poco más complejo que el resto, nos ofrece todas las características necesarias para cumplir con los requisitos del problema y aportar mayor seguridad en su uso.

Sistema de control

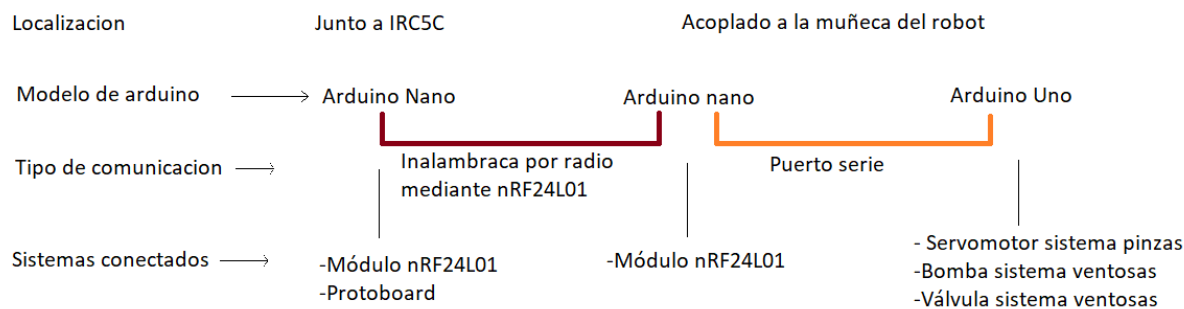
Como ya se ha mencionado en el apartado de soluciones posibles, se seleccionará el control del robot a través de su controlador IRC5C ya que es el único método disponible para ello.

En cuanto al control de las actuaciones de la herramienta, se ha seleccionado el tercer método expuesto: Sistema mixto con posibilidad de control por operario y robot + conjunto de placas Arduino.

Esta elección responde a la tendencia de la industria actual hacia una mayor automatización de los procesos a la par de la necesidad alcanzar mayor flexibilidad.

Como se ha expuesto dentro de las posibilidades estudiadas, el control de los actuadores se llevará a cabo a través varias placas Arduino interconectadas entre sí y a los actuadores y complementos necesarios.

Tras el estudio del conjunto de actuadores y sensores que será necesario acoplar, se ha seleccionado la forma de conexión entre ellos y las distintas placas Arduino (i2c, puerto serie...) como se mostrará en el apartado de resultados. Con ello presente y atendiendo a los puertos de comunicación restantes disponibles se ha seleccionado que los sistemas involucrados en el conjunto de control y su conexión entre ellos sean las siguientes [Esquema1]:



Esquema 1. Diagrama de conexiones entre sistemas Arduino

A su vez, ambos módulos nRF24L01 se han conectado a sus correspondientes placas Arduino a través del puerto i2c.

Se ha incluido otro Arduino Uno para el proceso de medición de distancia a través del sensor correspondiente y su muestra por pantalla. Ha sido necesario incluir un nuevo sistema Arduino ya que la comunicación del sensor de distancia con Arduino necesita del puerto serie o i2c para funcionar. En este caso, la pantalla ocupará el puerto i2c, por lo que la única opción es la de usar el puerto serie. Dicho puerto serie en el Arduino Uno encargado del control de la pinza y la ventosa ya se encuentra ocupado, puesto que es por donde recibirá las variables provenientes de la etapa anterior. Es por todo ello que ha sido necesario incluir otro Arduino Uno.

Todo el proceso de control será explicado en detalle dentro del punto 4 de desarrollo y resultados.

Actuaciones del sistema

Se han estudiado en los puntos anteriores las actuaciones que se deberán llevar a cabo dentro de la herramienta (manipulación mediante pinza o ventosas y medición de distancias), así como los elementos disponibles para ello. A modo de resumen se muestran a continuación los distintos sistemas encargados de cada una de las actuaciones dentro de la solución seleccionada:

- Servomotor: llevará a cabo el movimiento de cierre del sistema de la pinza. Conectado a su correspondiente Arduino mediante Shield 5.0.
- Bomba y válvula del sistema de ventosas: llevaran a cabo la actuación necesaria para accionar el método de manipulación de elementos mediante ventosas. Conectados a su correspondiente Arduino mediante Shield 5.0.
- Sensor de distancia: ofrecerá información sobre la distancia a objetos próximos. Conectado a su correspondiente Arduino mediante puerto serie.
- Pantalla LCD: mostrará los datos obtenidos a través del sensor de distancia. Conectado a su correspondiente Arduino mediante puerto i2c.

3.4.1. ELEMENTOS FINALES QUE COMPONEN MI DISEÑO

En este punto se muestra una tabla resumen de todos los elementos que estarán involucrados dentro del sistema que forma parte de este proyecto. Estos elementos se desarrollarán individualmente dentro del punto de desarrollo y resultados. [Tabla 10]

COMPONENTE	UNIDADES
Piezas mediante impresión 3D que forman la herramienta física	17
Controlador IRC5C del robot	1
Arduino Uno	2
Arduino Nano	2
Sensor Shield v5.0	1
Módulo de comunicación NRF24L01	2
Servomotor. [<i>Modelo Power HD - Servomotor Standard - HD-3001HB</i>]	1
Sistema de ventosas con bomba de succión. [Kit para 25T Servos MG996 MG995 DS3218]	1
Pantalla Lcd y módulo de comunicación I2C	1
Sensor de distancia VL53L0X	1
ProtoBoard	1
Interruptores	6
Led	6
Alimentación	1
Cableado de conexión	

Tabla 10. Elementos del sistema



Diseño y fabricación de una herramienta multifunción para un robot industrial.

4. DESARROLLO Y RESULTADOS

En este punto se detallará el desarrollo y resultado final de la herramienta. Para ello se partirá de la solución seleccionada anteriormente y se desarrollará cada punto específico del sistema hasta alcanzar su resultado final.

Los pasos que se han seguido para alcanzar un resultado satisfactorio dentro de este proyecto han sido:

- Diseño de los elementos mecánicos que componen la herramienta: se estudian y seleccionan las geometrías de cada uno de los componentes que formarán parte de la herramienta [1].
- Verificación de los elementos diseñados: se estudiará a través del software SolidWorks si todos los diseños son satisfactorios [2].
- Fabricación de los elementos diseñados mediante impresión 3D: haciendo uso de la tecnología de impresión 3D se conseguirán los modelos físicos de todos los elementos.
- Programación de las funcionalidades de la herramienta: se realizará la programación necesaria para que el conjunto del sistema sea capaz de funcionar de acuerdo con los requisitos de diseño
- Simulación del sistema mediante RobotStudio: se realizará una simulación del sistema completo a fin de verificar su correcto funcionamiento.
- Puesta en marcha del sistema real: una vez verificados todos los modelos simulados, se implementará el modelo real en el robot de la Universidad de Almería y se comprobará que su funcionamiento es satisfactorio.

4.1. PROCESO DE DISEÑO DE LOS ELEMENTOS MECÁNICOS

Una vez seleccionada la solución óptima para abordar el problema, se han llevado a cabo los distintos pasos que se exponen en este punto hasta alcanzar el diseño final de los elementos mecánicos que forman la herramienta.

Para alcanzar un correcto diseño de todos los elementos se han seguido los siguientes puntos:

- Se ha estudiado la geometría básica que ha de tener nuestra herramienta. Esto incluye los siguientes procesos:
 - Selección de la geometría de la base.
 - Verificación de que el sistema de 4 barras que formará la pinza cumple con todos los requisitos de diseño necesarios.
 - Selección de los engranajes necesarios para nuestro sistema.

- Selección de los puntos donde irán anclados los mecanismos de 4 barras y sus dimensiones para formar el sistema de pinza necesario.
- Se han diseñado cada una de las piezas que compondrán la herramienta para ser impresas posteriormente en 3D y así obtener el modelo físico.
- Se ha estudiado el ensamblaje resultante de la herramienta de manera que se compruebe que cumple con todos los requisitos de diseño.

4.1.1. GEOMETRIA BASICA DEL SISTEMA PINZA-VENTOSA

Como ya se ha expuesto anteriormente, dentro de este punto se muestra el resultado de las dimensiones fundamentales de la herramienta: se desarrolla desde la selección de las dimensiones de la base de las herramientas y sus correspondientes engranajes, hasta alcanzar las dimensiones necesarias de los elementos que formarán el sistema de la pinza dentro de la herramienta. Para ello se han utilizado las distintas metodologías estudiadas a lo largo del máster de ingeniería.

4.1.1.1. GEOMETRÍA DE LA BASE DE LA HERRAMIENTA.

En primer lugar, se ha seleccionado la geometría de la base de la herramienta. Estas dimensiones han sido seleccionadas para obtener un diseño óptimo atendiendo a los siguientes criterios:

- Se le ha seleccionado una anchura suficiente para el correcto ensamble del servomotor y los engranajes que harán posible el movimiento de las pinzas.
- Se le proporcionará una cara plana de 36 mm de anchura en el frontal de la herramienta para poder acoplar la herramienta ventosa que tiene un ancho de 20 mm. Con el aumento de esta cota también permitiremos un cierre correcto de las pinzas sin interferir con las ventosas.
- Se le ha dado una longitud de 120 mm, adecuada al alcance que se desea conseguir una vez la herramienta se encuentre completamente ensamblada, contando con la longitud que le proporcionan los brazos de la pinza y las ventosas.
- Se han tenido en cuenta las dimensiones del robot ABB IRB140 y su terminal donde irá anclada la herramienta para conseguir una buena correlación entre sus dimensiones.
- Se le han realizado 2 chaflanes en la parte frontal de la herramienta para un correcto acople de las pinzas que posteriormente se incluirán.

La geometría resultante es la que se muestra a continuación [Figura 4.1]:

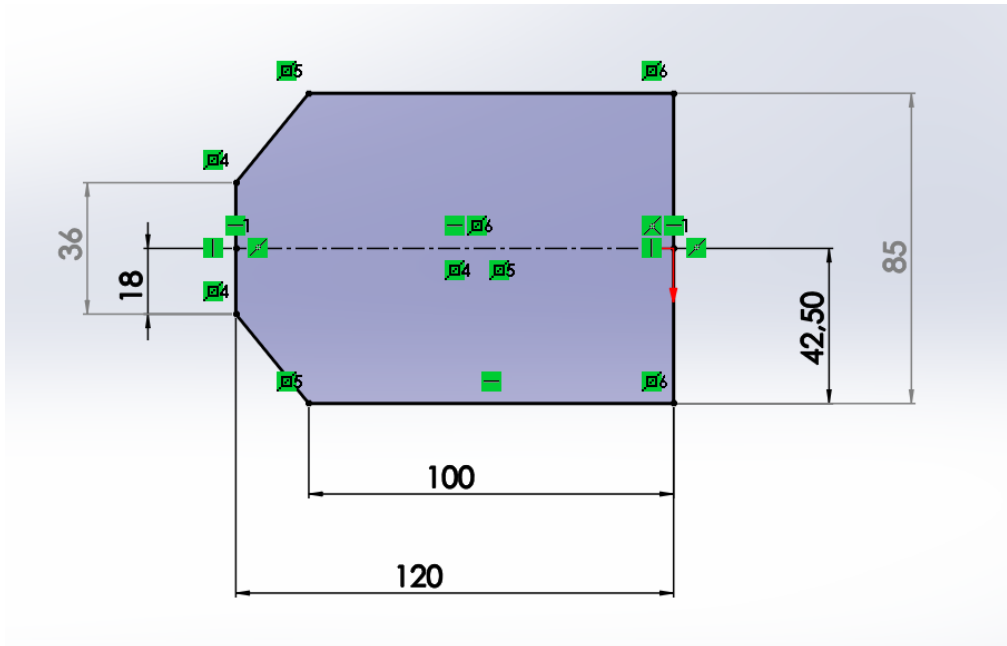


Figura 4. 1 Croquis inicial de la base de la herramienta

Sobre esta geometría se aplicará posteriormente el diseño mediante SolidWorks para alcanzar la pieza correspondiente a la base de la herramienta final.

4.1.1.2. REQUISITOS A CUMPLIR POR EL MECANISMO DE 4 BARRAS DE LA PINZA

Una vez obtenida la geometría que formará la base, se ha de llevar a cabo el estudio de las dimensiones que tendrán las pinzas de la herramienta.

Estas pinzas estarán formadas por un mecanismo de 4 barras que ha de tener 1 grado de libertad, ya que, dado el giro del eslabón accionado por el servomotor, la pinza ha de tener una posición definida.

Se ha seleccionado un mecanismo de 4 barras de 1gdl como el que se muestra en la figura [Figura 4.2]:

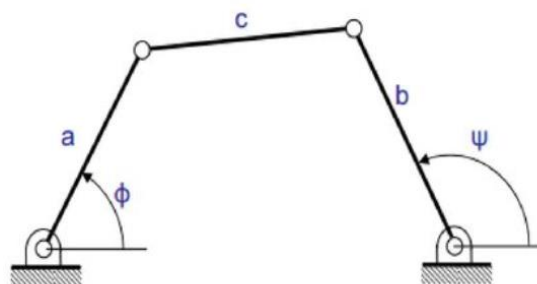


Figura 4. 2. Mecanismo 4 barras genérico

Se ha realizado la comprobación de que nuestro mecanismo tiene 1 grado de libertad y cumple la ecuación de síntesis de número. Esto se puede observar dentro del Anexo1 de cálculo en el punto “1. SINTESIS DE NÚMERO DE UN MECANISMO DE 4 BARRAS”.

Debido a la geometría de nuestro problema, también es necesario que los terminales de la pinza se encuentren siempre de manera paralela al eje longitudinal de la misma [Figura 4.3]. Con ello se conseguirá un contacto completo entre el terminal de la pinza y la pieza a manipular.

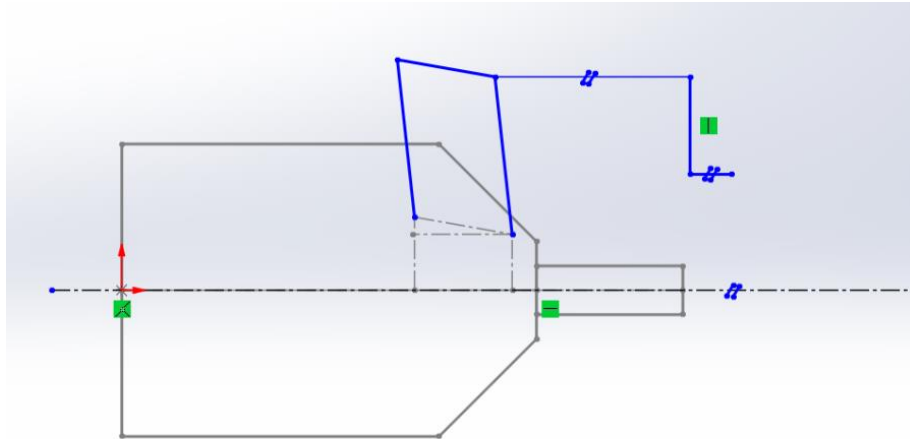


Figura 4. 3.Paralelismo entre elementos de la pinza y eje de la herramienta

Para conseguir esta simetría, mediante el estudio de la síntesis dimensional del mecanismo que se puede encontrar en el anexo 1 de cálculos en el punto “2. SÍNTESIS DIMENSIONAL DEL MECANISMO DE LA PINZA”, se ha determinado que los eslabones a y b han de tener la misma longitud, al igual que los eslabones c y d. Esto permitirá que se mantenga el paralelismo entre estos eslabones de igual dimensión, con lo que realizando un correcto diseño de el eslabón c donde se incorpora el saliente que formará la pinza, se conseguirá que la cara que va a quedar en contacto con el elemento a manipular siempre esté paralela al eje longitudinal de nuestra herramienta.

4.1.1.3. SISTEMA DE ENGRANAJES

El sistema de la pinza estará formado por dos mecanismos de 4 barras simétricos respecto al eje longitudinal de la herramienta. Uno de estos mecanismos estará directamente accionado por el servomotor a través del acoplamiento de 2 engranajes:

- Engranaje acoplado directamente al servomotor
- Engranaje que formará parte del eslabón del mecanismo de 4 barras

De esta manera el movimiento del servomotor accionará el mecanismo de pinza.

A su vez, los dos mecanismos de 4 barras estarán conectados a través de sus piezas A1 y A2 como puede observarse en la siguiente imagen [Figura 4.4]:

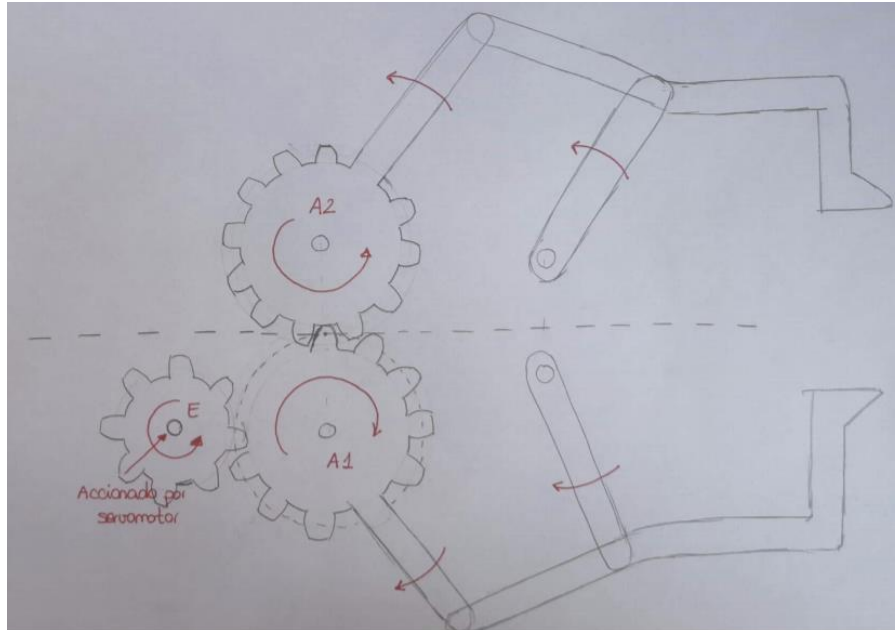


Figura 4. 4.Croquis de conexiones entre engranajes del mecanismo de 4 barras para la pinza

De esta manera los dos mecanismos actuarán de manera conjunta cerrando y abriendo la pinza. La pieza E conectada directamente al servomotor, transmitirá su movimiento de giro al eslabón A1, y esta a su vez al eslabón simétrico A2 (referencias a nomenclatura en figura 4.4).

El requisito fundamental que se ha de tener en cuenta para el diseño de los engranajes es que el movimiento de giro proporcionado por el servomotor se transforme en el movimiento de nuestro sistema de 4 barras. Para ello ha de convertirse el ángulo proporcionado por el servomotor de 180º, en el movimiento de que permita abrir y cerrar la pinza un total de 90º. Para ello, la relación de transmisión entre el engranaje del servomotor (entrada) y el del eslabón del mecanismo de 4 barras (salida) ha de ser de ½.

Tras el estudio del sistema de engranajes necesario, se obtiene la geometría a implementar [Tabla 11]:

	ESLABON DE ENTRADA	ESLABONES DE SALIDA
DIAMETRO DE PASO	24	48
NUMERO DE DIENTES	12	24
MODULO	2	2

Tabla 11. Resumen de parámetros de los engranajes

El proceso de obtención del sistema de engranajes necesario puede consultarse dentro del anexo1 de cálculos en el punto “3: DIMENSIONAMIENTO DEL SISTEMA DE ENGRANAJES”.

Con los datos calculados, podremos obtener los engranajes correspondientes normalizados a través de la biblioteca de SolidWorks:

- Engranaje de entrada [Figura 4.5]:

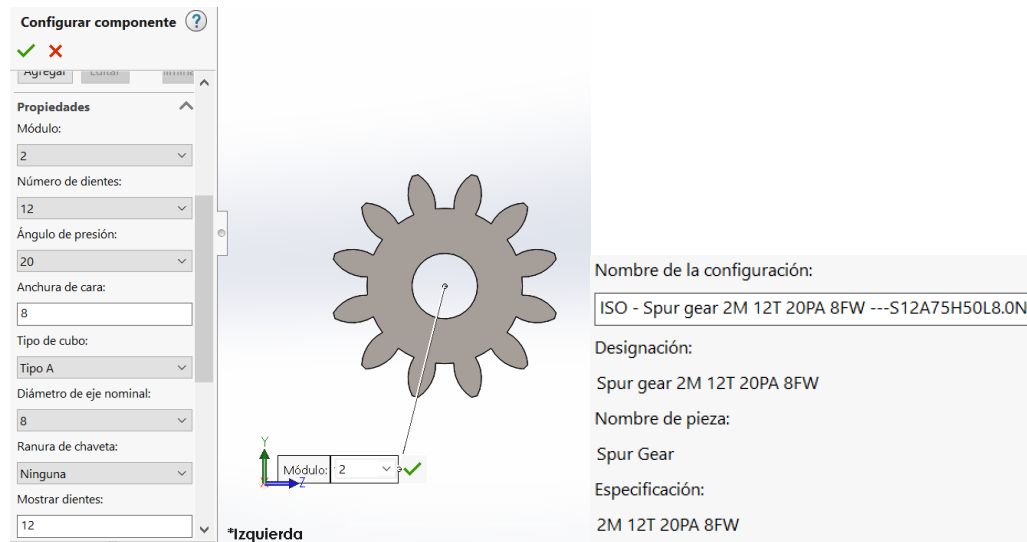


Figura 4. 5. Engranaje de transmisión de potencia desde el servomotor hacia el mecanismo de la pinza

- Engranaje de salida [Figura 4.6]:

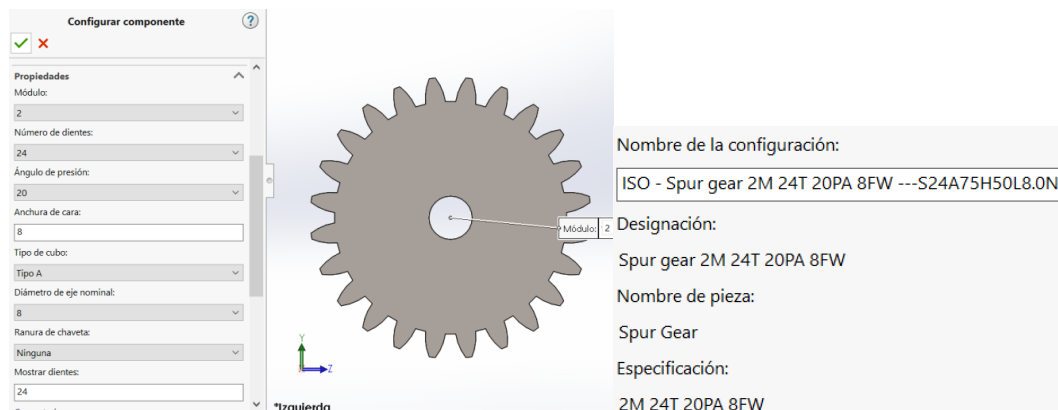


Figura 4. 6. Engranaje de transmisión de potencia para la actuación de la pinza

Los datos correspondientes al ancho del engranaje y al eje central se incorporarán posteriormente de acuerdo con el diseño de la herramienta completa.

4.1.1.4. PUNTOS DE ANCLAJE Y DIMENSIONES DEL MECANISMO DE 4 BARRAS

Del proceso desarrollado anteriormente conocemos los siguientes requisitos de diseño:

- Dimensión de la base seleccionada.
- Mecanismo de 4 barras con 1 gdl y sus lados $a=b$ y $c=d$.
- Eslabón de contacto con la pieza a manipular paralelo al eje longitudinal

A estos requisitos han de sumarse también los siguientes:

- En el punto de cierre de la pinza el alcance de la herramienta ha de ser de al menos 100 mm (doble del ancho de la pieza máxima a manipular)
- En el punto de máxima apertura de la pinza, el mecanismo 4 barras debe tener la geometría necesaria para no interferir con la herramienta ventosa. De esta manera se permitirá un uso de ambas herramientas sin interferencias.
- La pinza en posición de máxima apertura ha de tener una distancia suficiente para manipular piezas de hasta 100 mm de ancho.

Con estos requisitos y conociendo el sistema de engranajes mediante el cual se accionará el sistema, se seleccionarán las dimensiones del mecanismo de 4 barras y los puntos de unión de este sistema con la base de la herramienta.

Para llevar a cabo este diseño se ha seguido el siguiente proceso mediante métodos gráficos:

1. Selección de los puntos de anclaje a la base.

Conociendo las dimensiones de la base de la herramienta y que el eslabón motor del sistema de 4 barras irá acoplado a un engranaje de 48 mm de diámetro, coloco los puntos A0 y B0 (anclajes de las barras a y b) dentro del croquis de la base según se muestra en la Figura 4.7.

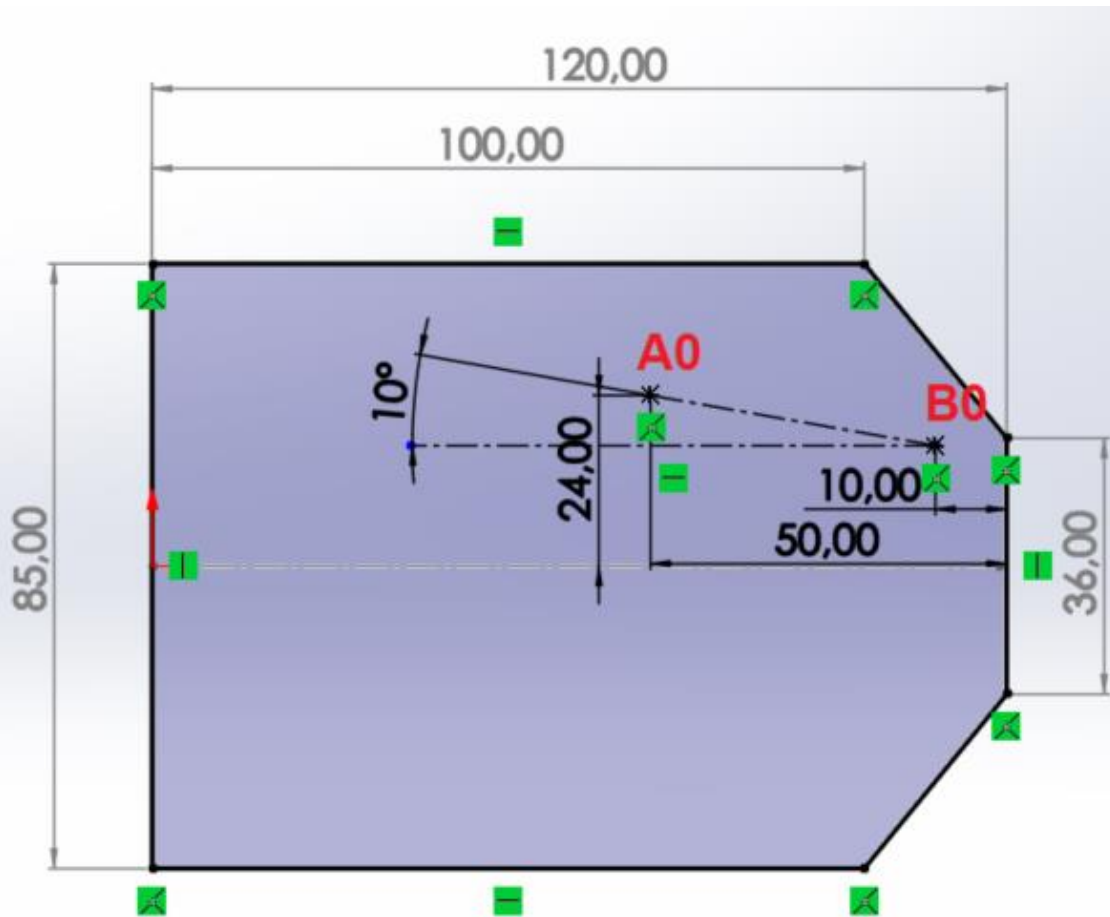


Figura 4. 7. Croquis de los puntos de anclaje a la base de los elementos del sistema de 4 barras

- Se ha colocado el punto A₀ a 24 mm de distancia del eje longitudinal de la base para realizar el acoplamiento entre los dos mecanismos simétricos de 4 barras que formarán la pinza, ya que el radio nominal de los engranajes de entrada de potencia tiene esa medida. De esta forma se transmitirá el movimiento entre ellos.
- El ángulo entre los puntos A₀ y B₀ será de 10° y posteriormente se deberá tener en cuenta para el diseño del eslabón c.
- Las distancias de A₀ y B₀ al extremo frontal de la base se han seleccionado tras un proceso de iteración con distintas medidas para un mejor diseño.

El acoplamiento de las circunferencias base de ambos engranajes será como se representa en la siguiente figura 4.8:

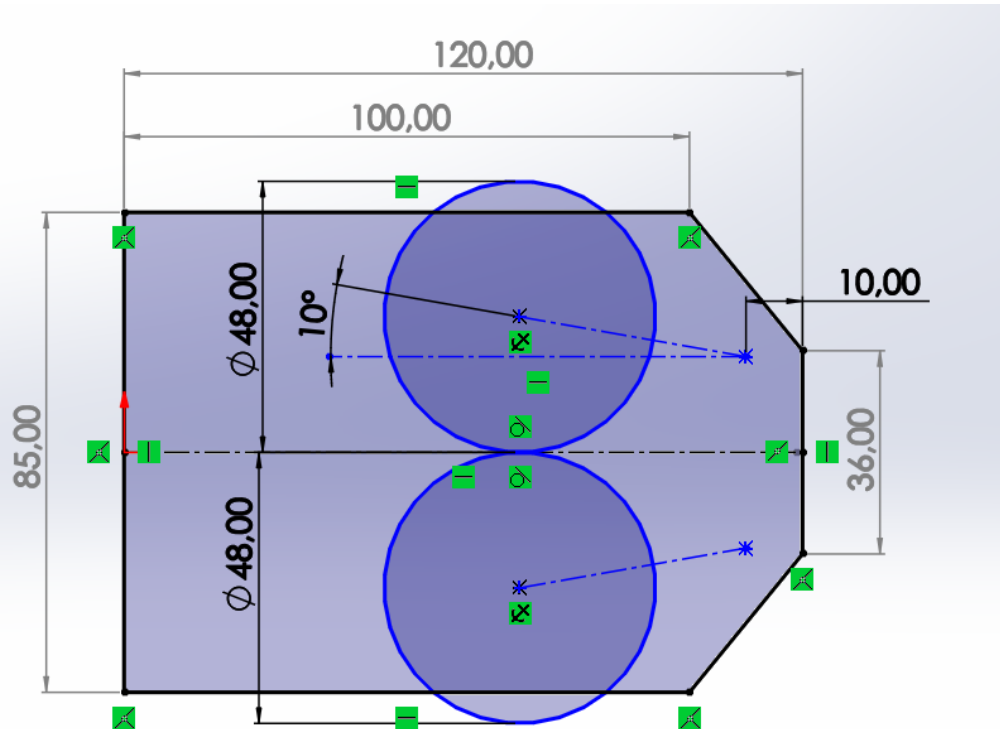


Figura 4. 8. Acoplamiento de las circunferencias base de ambos engranajes

De esta manera la medida del eslabón d ya es un dato conocido ($d=40,62$ mm) [Figura 4.9]

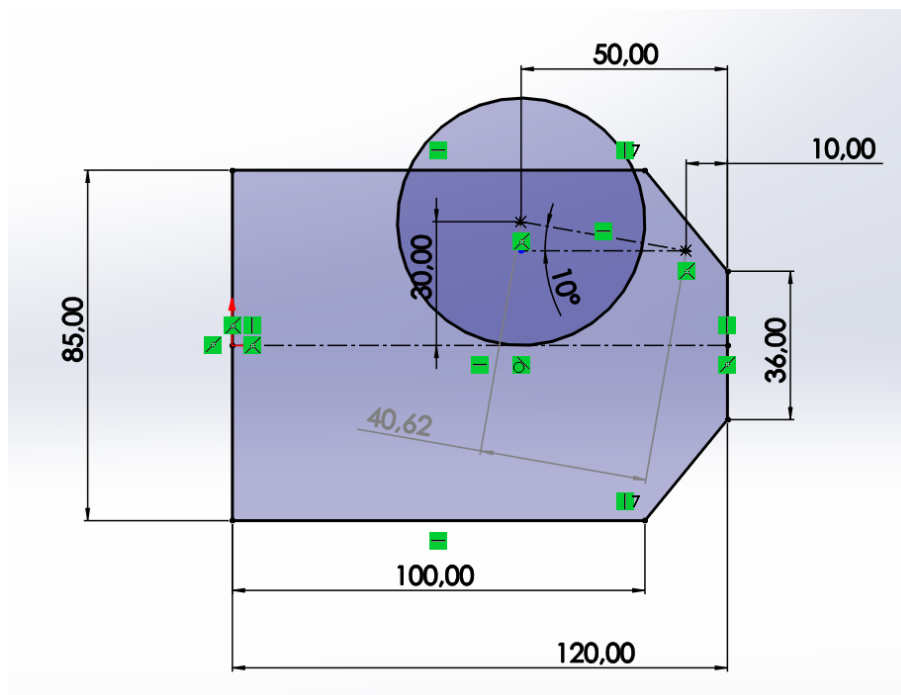


Figura 4. 9. Detalle de posición del engranaje y medida eslabón d en el sistema de pinza

2. Eslabones del sistema de 4 barras

Partiendo de los puntos de anclaje seleccionados anteriormente y tras un proceso de diseño gráfico iterativo, se ha alcanzado una solución que satisface todos los requisitos de diseño [Figura 4.10], la cual se explicará mediante el desarrollo de este apartado.

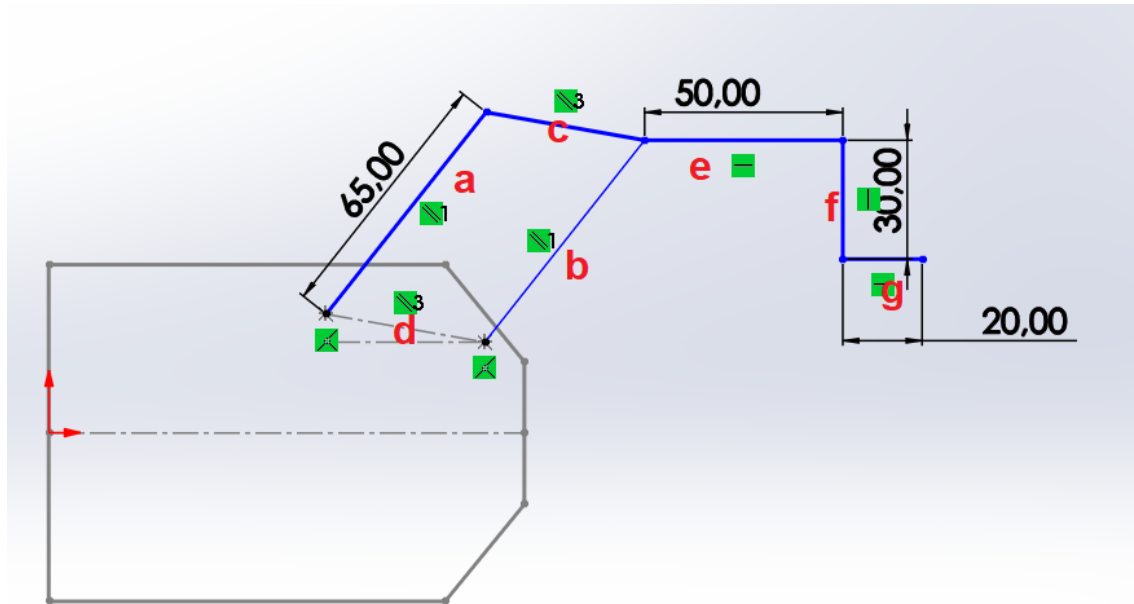


Figura 4. 10. Dimensión de los eslabones del sistema de 4 barras

La nomenclatura mostrada en la anterior imagen será la que se siga durante este apartado para una mejor explicación del proceso de selección de las distintas medidas.

Cabe recordar que, en nuestro diseño, las longitudes de los lados paralelos serán las mismas ($a=b$ y $c=d$).

Para la prolongación del eslabón c que formará el cabezal de nuestra pinza se ha de respetar el paralelismo de los lados e y g con respecto al eje longitudinal, para de esta manera conseguir siempre el mayor contacto del tramo final g con la pieza a manipular. Esto se conseguirá respetando el ángulo de 10° existente entre el eslabón d y el plano longitudinal dentro del diseño de la nueva barra [Figura 4.11].

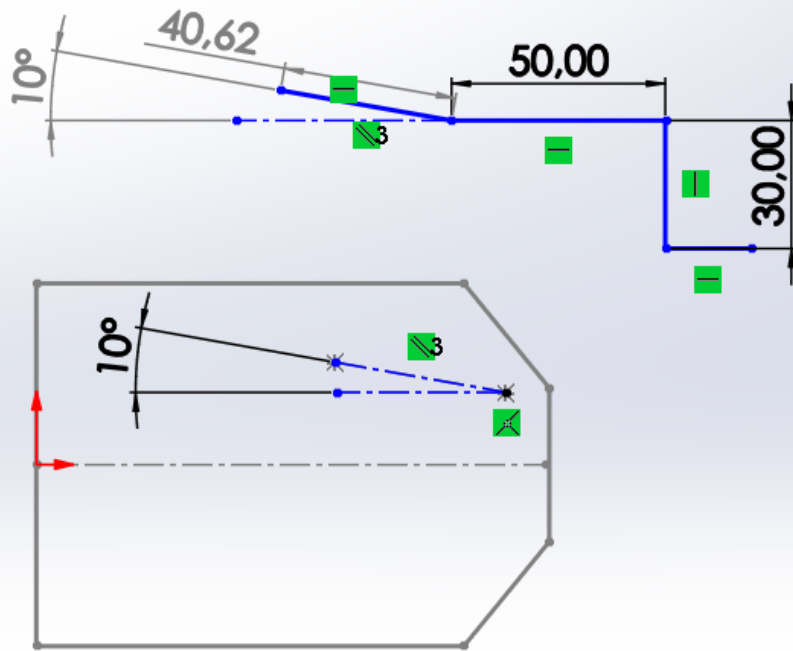


Figura 4. 11. Detalle de paralelismo entre eslabón final y eje de la herramienta

Para la selección de la distancia e , se ha tenido en cuenta que el alcance máximo que se desea obtener ha de ser como mínimo de 100 mm, es por ello por lo que se ha seleccionado 50 mm en este eslabón, ya que una vez extendida la pinza en la posición de cierre máximo nos proporcionará un alcance mayor a estos 100 mm [Figura 4.12].

Para la dimensión del eslabón g se ha tenido en cuenta disponer de una distancia suficiente para proporcionar un buen contacto entre la superficie de la pinza y la pieza a manipular. Se ha seleccionado 20 mm como un valor adecuado.

Posteriormente, para la selección de las dimensiones de las barras a y b se tendrá en cuenta que su altura nos proporcionará la longitud para hacer posible que el alcance sea superior a los 100 mm en el sentido del eje longitudinal cuando la pinza cierra al máximo (servo 0°) y superior a 100 mm en el eje transversal cuando la pinza se encuentra en su posición de máxima apertura (servo 180°). El valor del eslabón f se seleccionará de manera conjunta con los de a y b para permitir una apertura de la pinza de al menos 100 mm. [Figura4.12]

Teniendo esto en cuenta los valores seleccionados para a , b y f son:

$$a = b = 65 \text{ mm}$$

$$f = 30 \text{ mm}$$

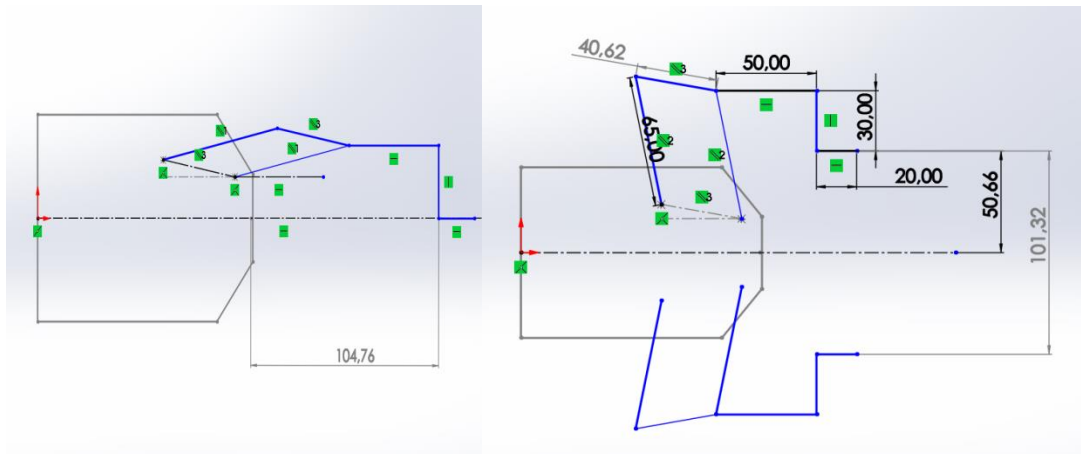


Figura 4. 12. Detalle medidas eslabones a, b y f

Serán necesarias también varias comprobaciones en las posiciones extremas de nuestro mecanismo; servomotor en 0° (máximo cierre) y servomotor en 180° (máxima apertura).

SERVOMOTOR EN 0° (MÁXIMO CIERRE) [Figura 4.13]:

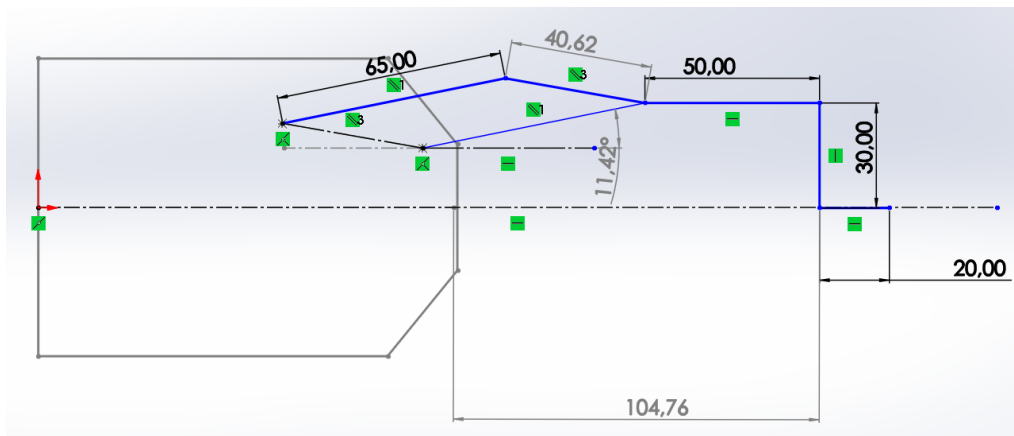


Figura 4. 13. Posición de cierre máximo del mecanismo de 4 barras

Se han realizado las siguientes comprobaciones:

- El ángulo existente entre el eslabón más cercano a la ventosa (b) y el eje longitudinal hace posible que no haya interferencia entre ambos métodos de manipulación.
- Verificamos que el alcance es mayor a 100 mm. Alcance = 104,76 mm.

SERVOMOTOR EN 180º (MÁXIMA APERTURA)

Se comprueba que en esta posición [Figura 4.14], el extremo de la pinza quedará por detrás de la herramienta ventosa, con lo que no existirá interferencia entre ambas herramientas, puesto que si se quiere utilizar la ventosa tan solo se ha de tener la precaución de llevar la pinza hasta la posición de máximo giro del servomotor:

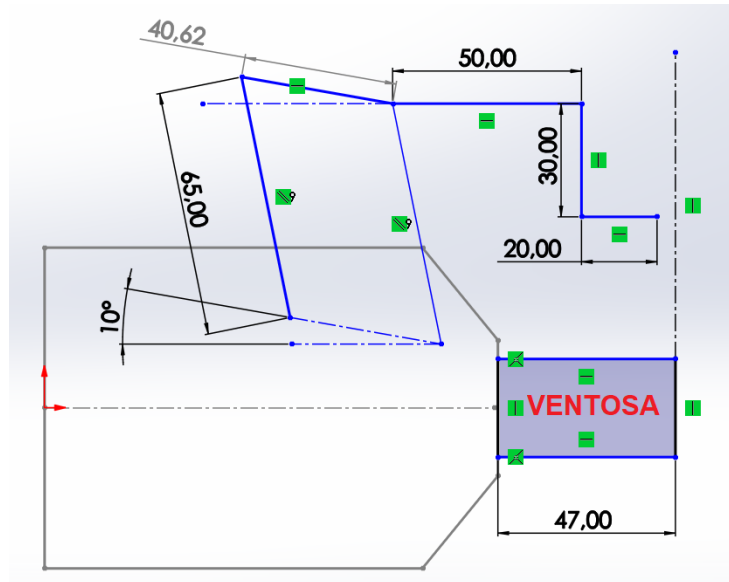


Figura 4. 14.Posición de apertura máxima del mecanismo de 4 barras

4.1.1.5. DIMENSIONES FINALES

De esta manera finalmente obtenemos las dimensiones de nuestra base junto con nuestro doble mecanismo de 4 barras que formará la pinza [Figura 4.15]:

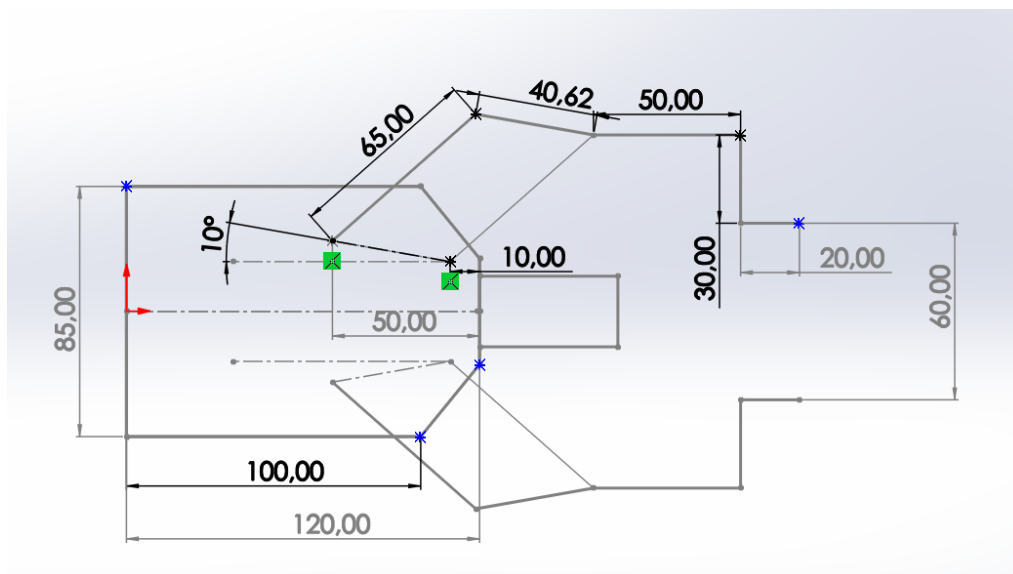


Figura 4. 15. Dimensiones finales mecanismo 4 barras para obtener el sistema de pinza. Ejemplo para una apertura de 60 mm

4.1.2. DISEÑO DE LAS PIEZAS FINALES QUE COMPONEN LA HERRAMIENTA.

Una vez conocemos las dimensiones que ha de tener el sistema de la pinza y la base donde ha de ir colocada, podremos diseñar uno a uno todos los elementos que van a componer nuestra herramienta. Las piezas cuyo diseño se muestra a continuación son las siguientes:

1. Engranaje de acople al servomotor.
2. Eslabones del sistema que formará la pinza.
 - Eslabón simple.
 - Eslabón con engranaje para acople al sistema de potencia.
 - Eslabón con terminal de pinza.
3. Base de la herramienta.
 - Base inferior.
 - Base superior.
4. Soportes para los elementos auxiliares necesarios.
 - Soporte superior.
 - Soporte inferior.
5. Carcasa para sistema de alimentación.
6. Soporte para sensor de distancia.

Todos los planos de las distintas piezas pueden encontrarse en el anexo de planos correspondiente a este TFM.

1. ENGRANAJE CON ACOPLA AL SERVOMOTOR

En primer lugar, se realizará el acople del engranaje de entrada diseñado previamente con el servomotor.

Para ello hemos de hacer una conexión entre las dos piezas siguientes:

- Engranaje de entrada [Figura 4.16]:

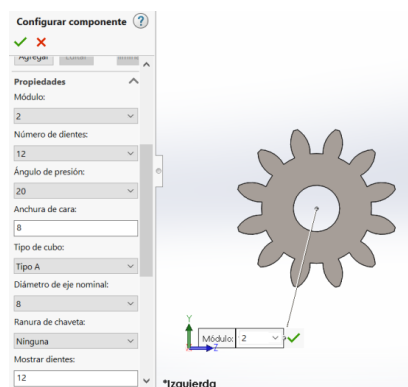


Figura 4. 16. Engranaje de la pieza acoplada al servomotor

- Pieza de acople del servomotor [Figura 4.17]:



Figura 4. 17. Pieza de acople del servomotor

La pieza de acople disponible en el servomotor tiene la siguiente geometría [Figura 4.18]:

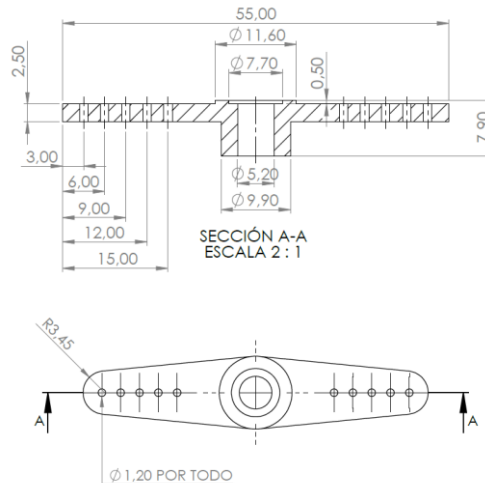


Figura 4. 18. Geometría pieza de acople del servomotor

Conociendo los dos elementos que se han de acoplar, se diseña la siguiente pieza [Figura 4.19]:

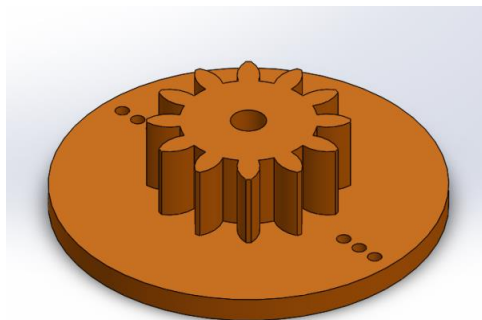


Figura 4. 19. Pieza de acople al servomotor

En ella se ha incluido el engranaje de entrada correspondiente y también una base plana en la que encontramos los taladros que nos permitirán el acople al servomotor.

La unión de ambas piezas podemos observarla en la siguiente imagen [Figura 4.20]:

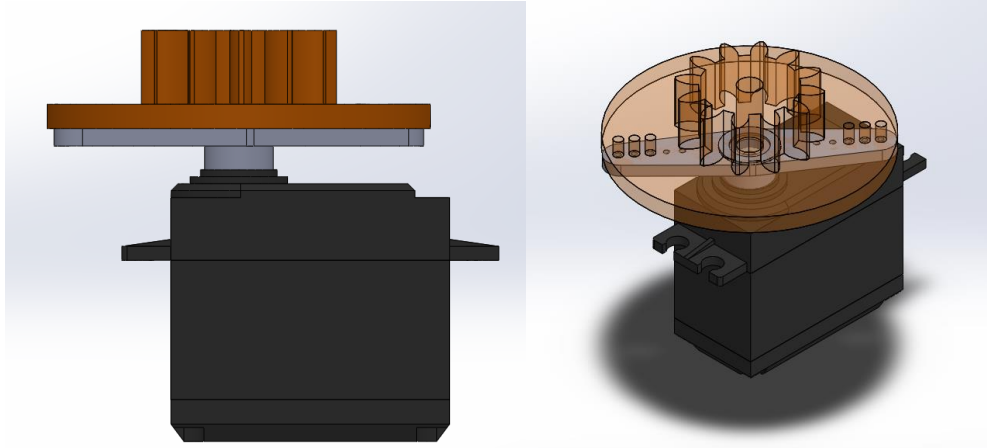


Figura 4. 20. Detalle acople al servomotor

La altura de esta pieza se ha diseñado teniendo en cuenta su acople a la base, el cual se verá en el punto siguiente.

2. ESLABONES DEL SISTEMA DE PINZA

Estos eslabones serán los correspondientes al diseño del mecanismo de 4 barras que se ha visto con anterioridad. Las medidas finales del mecanismo seleccionadas tras el proceso de diseño fueron las observadas en la siguiente figura 4.21:

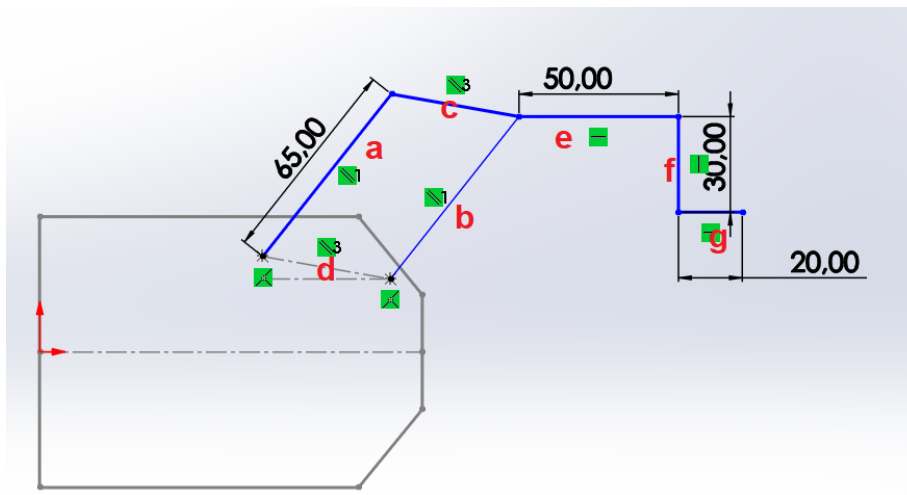


Figura 4. 21. Dimensiones finales mecanismo de 4 barras

Las barras diseñadas responderán a uno de los 3 tipos siguientes:

- **Eslabón con engranaje:** corresponderá a la barra *a* que va en la cara superior de la base como se verá a continuación [Figura 4.22].

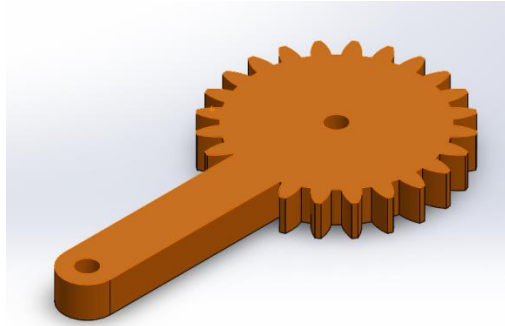


Figura 4. 22. Eslabón con engranaje

- **Eslabones simples:** *a* (sin engranaje) y *b* [Figura 4.23].

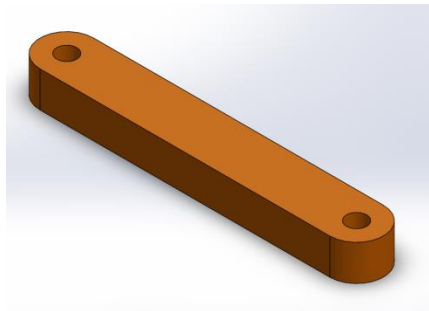


Figura 4. 23. Eslabón simple

- **Eslabón de pinza** [Figura 4.24]: Barra c-e-f-g.

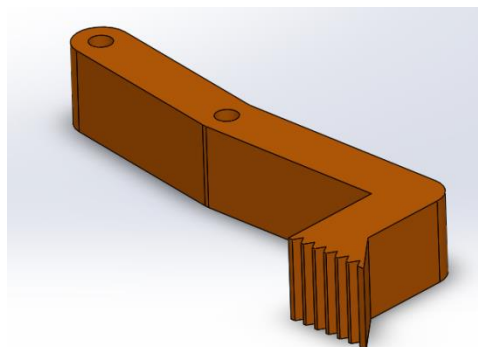


Figura 4. 24. Eslabón pinza

Se Incluye el diseño de un sistema de dientes en el extremo de la pinza para una mayor adherencia con la pieza a manipular.

Modificación del eslabón que incluye al engranaje:

Cabe destacar que, con el fin de que el engrane de las dos piezas simétricas sea correcto y sus eslabones se encuentren siempre en posiciones totalmente simétricas, se ha tenido que modificar la posición de los dientes de una de ellas, desplazándolos el ángulo existente entre el punto medio del valle y del diente [Figura 4.25]:

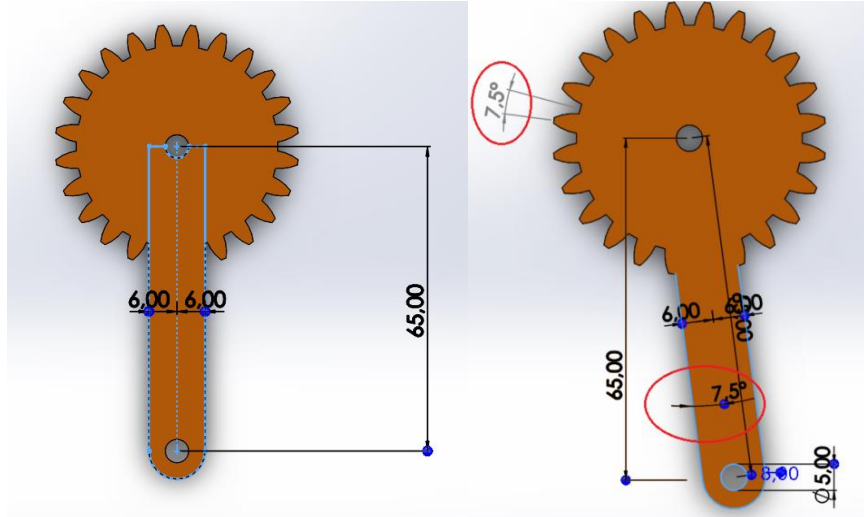


Figura 4. 25. Eslabón con engranaje modificado

De esta manera los engranajes coincidirán a la perfección y la orientación y posición de las pinzas de cada uno de los lados será totalmente simétrica [Figura 4.26]:

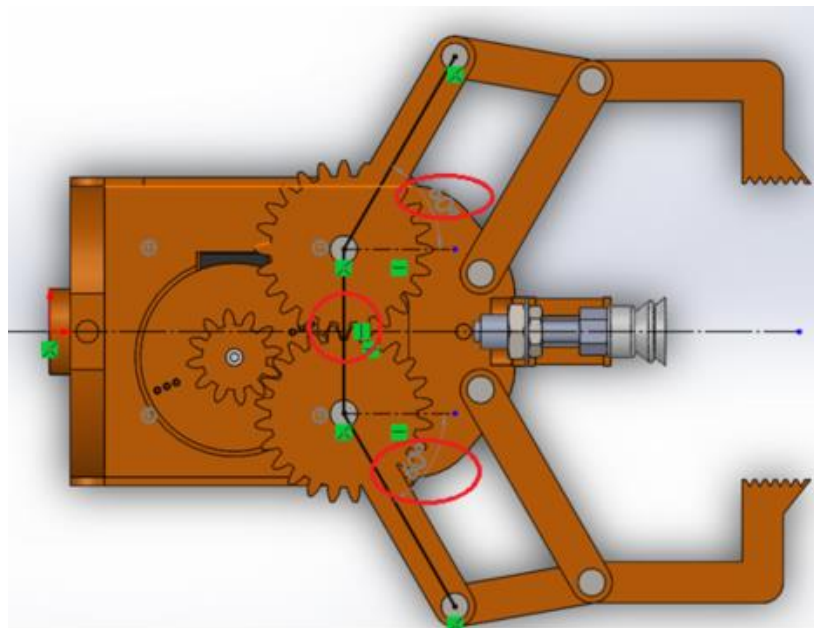


Figura 4. 26. Detalle acoplamiento entre engranajes del sistema pinza

Se puede verificar también esta condición en las dos posiciones más extremas del sistema:

- Completamente cerrado [Figura 4.27]:

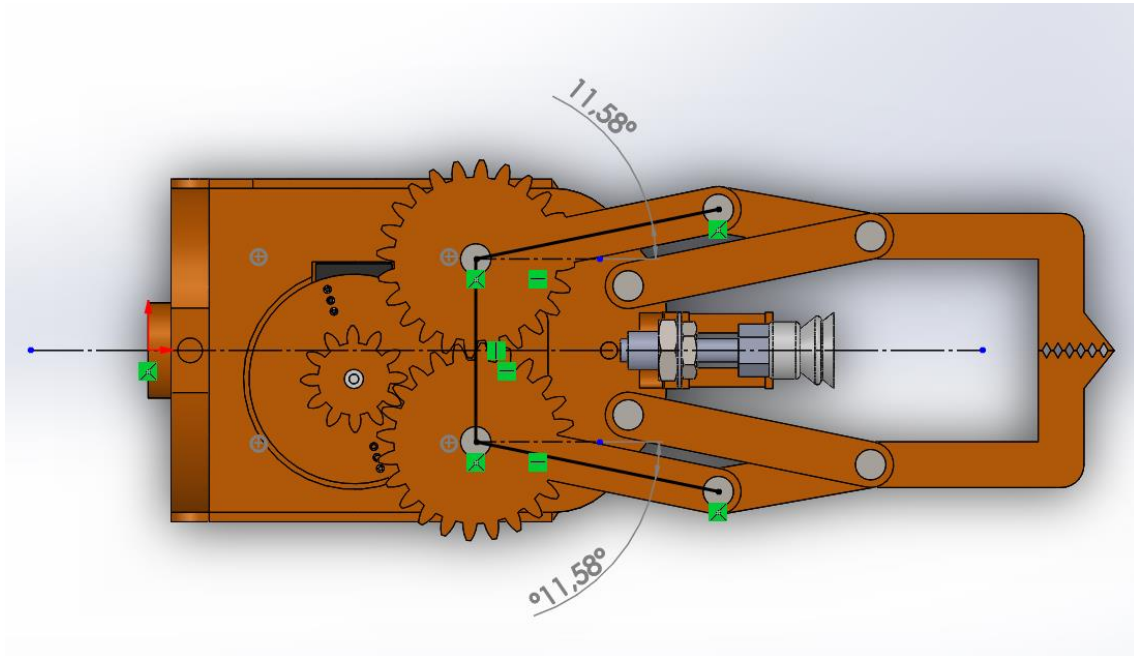


Figura 4. 27. Sistema de pinza totalmente cerrado

- Completamente abierto (+90°) [Figura 4.28]:

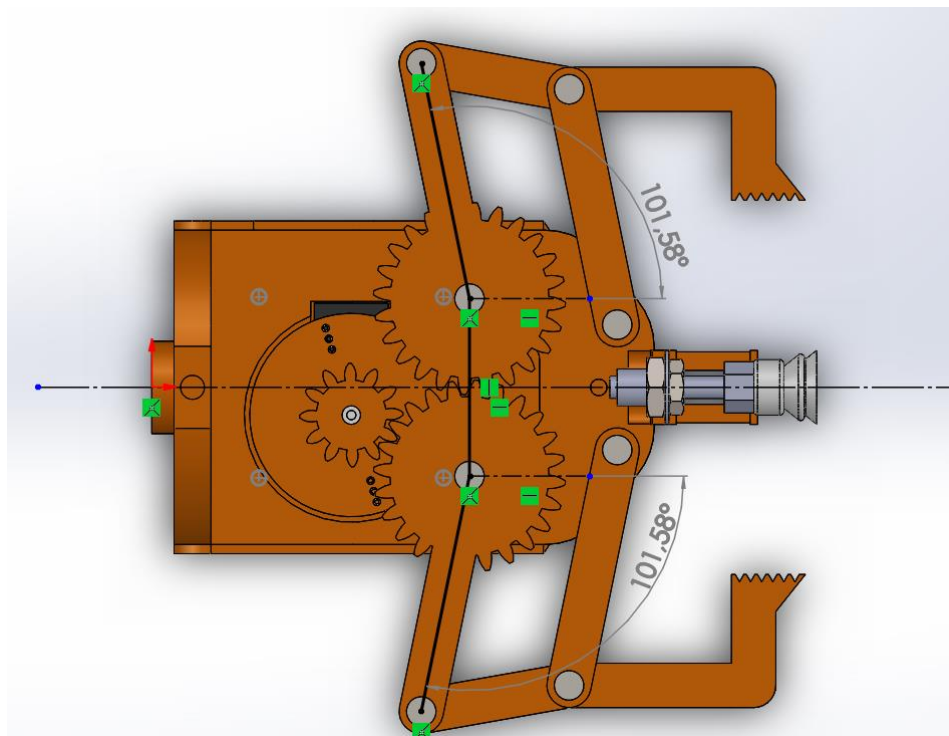


Figura 4. 28. Sistema de pinza totalmente abierto

Ensamble de los componentes que compondrán el sistema de la pinza:

Se incluye en este punto las imágenes que incluyen la base de la herramienta para una mejor comprensión del sistema. Esta se desarrollará en el siguiente punto.

Con el fin de que el plano medio de la pinza coincida con el de la base de la herramienta y a su vez, este contenga al eje longitudinal de la herramienta, se diseña el sistema de modo que el eslabón c-e-f-g que forma la pinza, tenga la misma altura que la base y quede unida a ella por sus caras inferior y superior de la siguiente manera [Figura 4.29]:

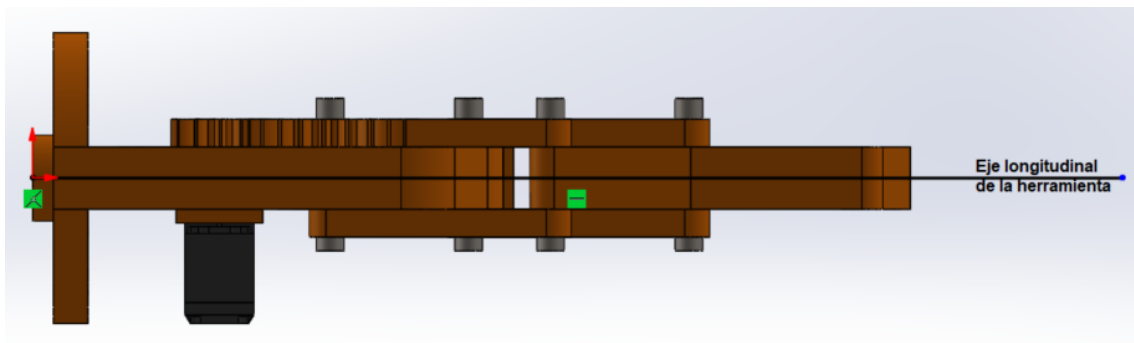


Figura 4. 29. Detalle ensamblado del sistema de pinzas

El resultado del sistema de pinzas de manera conjunta con la base el mostrado en las siguientes figuras 4.30 y 4.31:

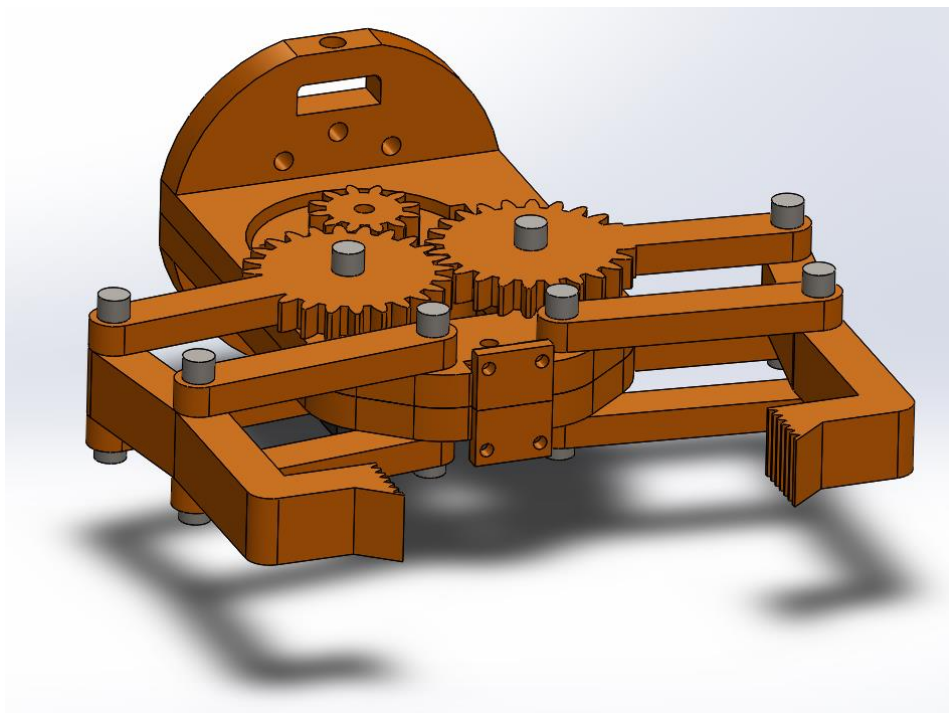


Figura 4. 30. Resultado del sistema de pinzas. Vista frontal

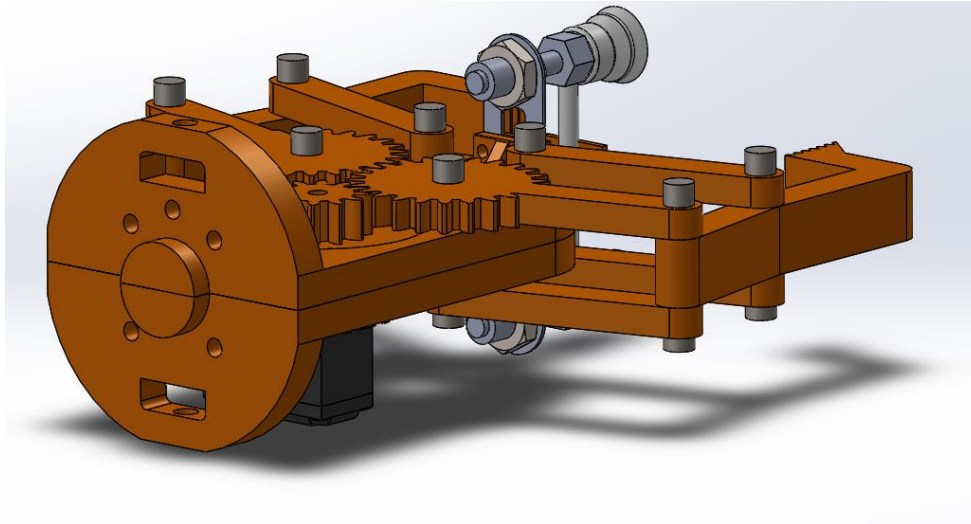


Figura 4. 31. Resultado del sistema de pinzas. Vista posterior

De esta forma, cada lado simétrico de la pinza estará formada por:

- 3 eslabones básicos de 65 mm
- 1 eslabón de 65 mm con engranaje
- 1 eslabón c-e-f-g que forma la pinza.

3. BASE DE LA HERRAMIENTA

Dentro de la base de la herramienta se han de tener en cuenta una serie de elementos para su posterior acople. Estos elementos son:

- Servomotor
- Mecanismo de 4 barras de la pinza
- Sistema de ventosas
- Sensor de distancia

Otras consideraciones de diseño:

- Se ha de tener también en cuenta que se le añadirá a nuestro diseño dos soportes (uno por la parte interior y otro por la superior) para la sujeción de los demás elementos que componen el sistema.
- Se ha de diseñar el sistema de acople al robot ABB IRB140 con el que trabajará la herramienta.

Con todo ello, en primer lugar, se estudia el acoplamiento del servomotor a la base. Se decide dividir la base en 2 piezas para hacer posible un correcto acople del sistema del servomotor.

Las 2 piezas que formarán la base son las expuestas a continuación:

Base superior [Figura 4.32]

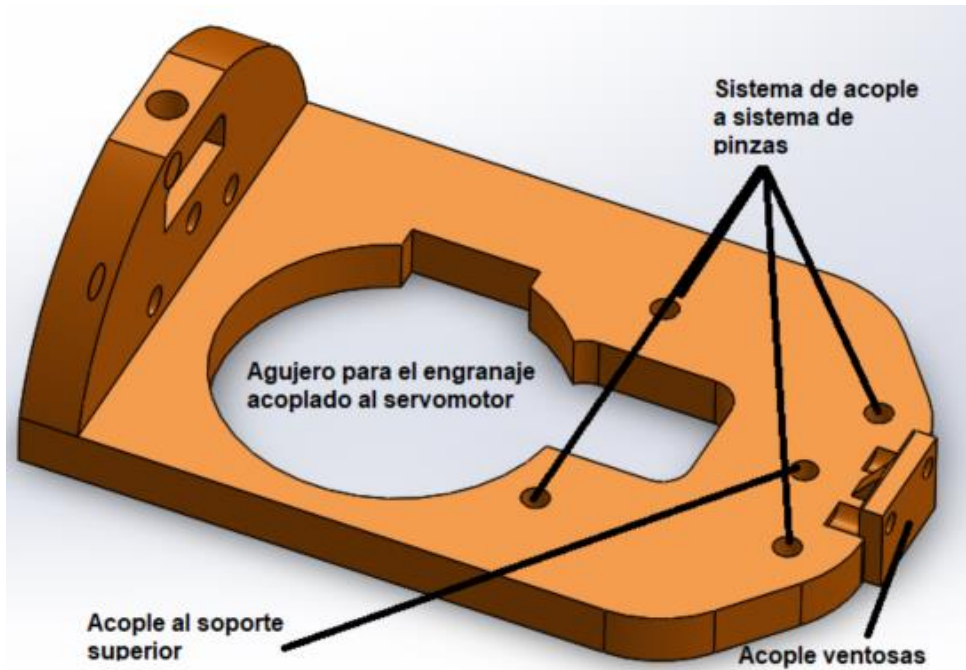


Figura 4. 32. Base superior y características.

Base inferior [Figura 4.33]

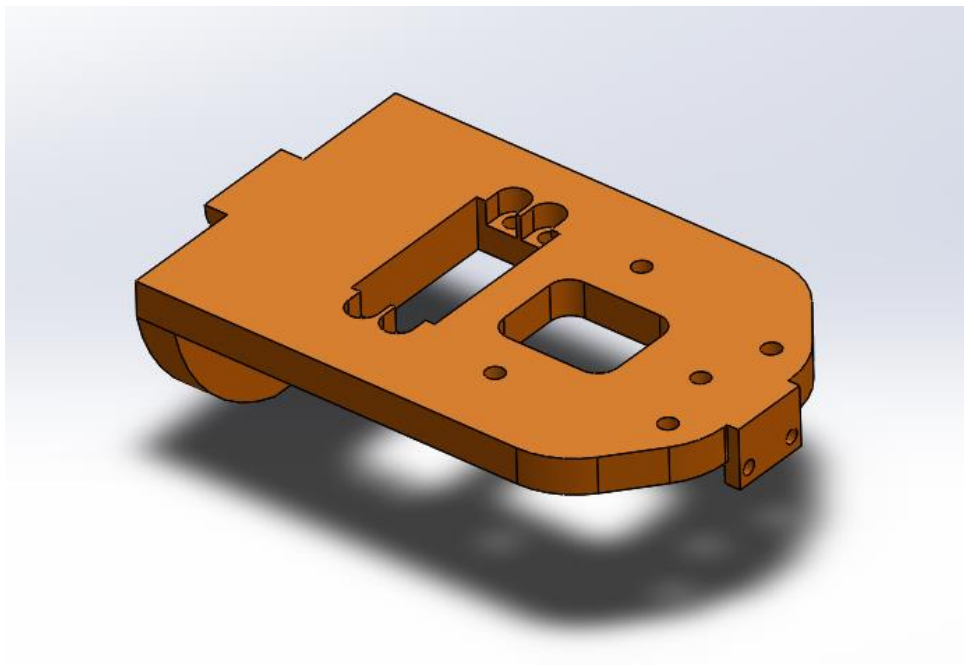


Figura 4. 33. Base inferior

Cuyas principales características son [Figura 4.34]:

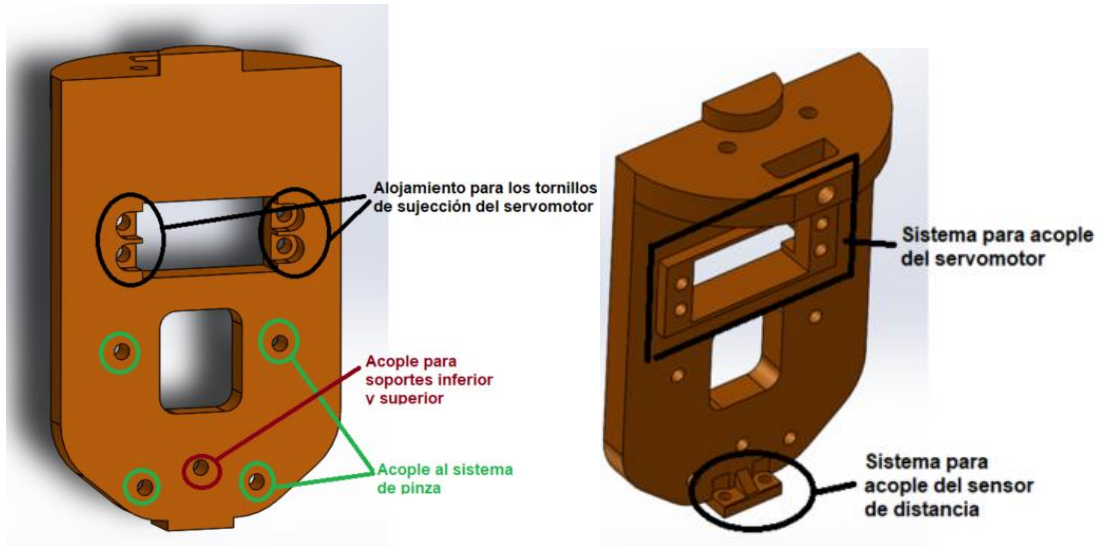


Figura 4. 34. Detalles base inferior

Base completa:

Con el diseño de este acople al servomotor, se permite una correcta unión entre el servo y la base, de manera que el engranaje que funciona como entrada de potencia al sistema de pinza quede a la altura correcta para que las pinzas se desplacen tangentes a la cara superior de la base [Figura 4.35]:

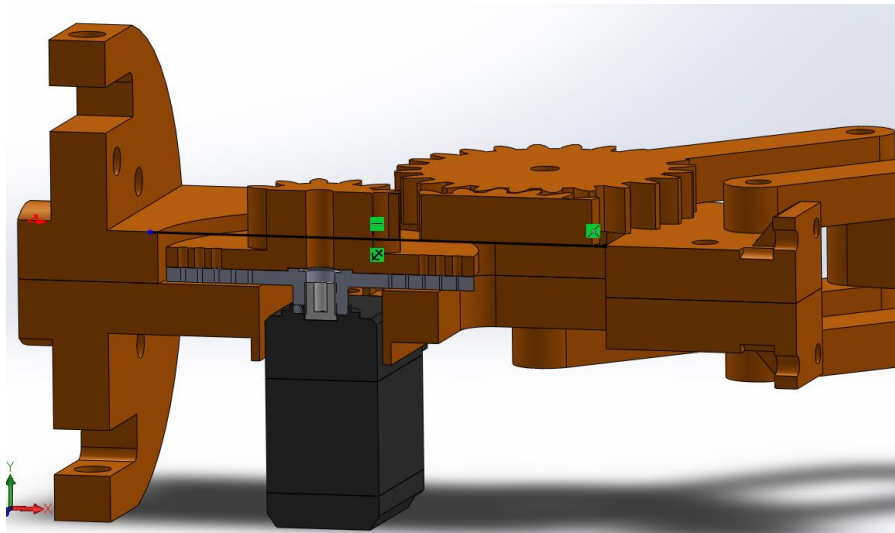


Figura 4. 35. Corte base completa por el plano de acople al servomotor

Se permite también libertad en el movimiento de giro del servomotor gracias a la apertura circular de la parte superior de la base [Figura 4.36]:

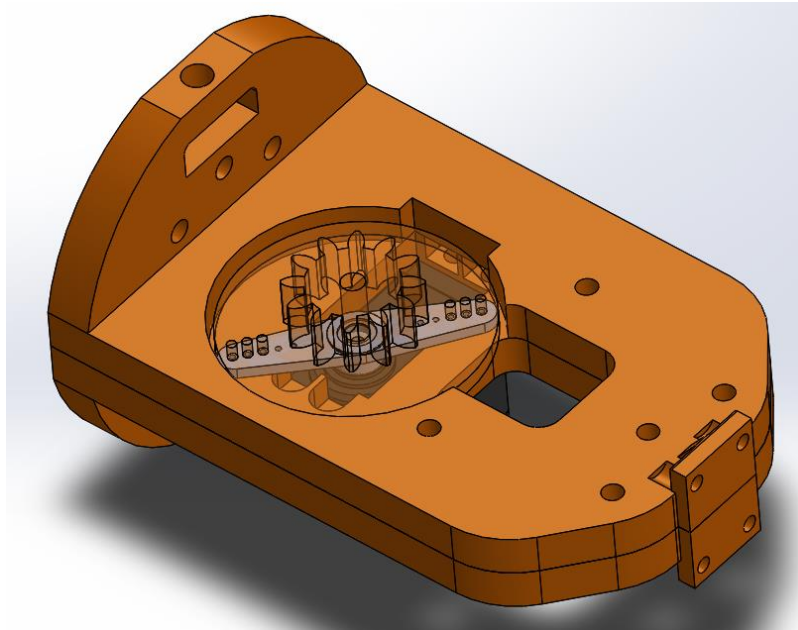


Figura 4. 36. Detalle para acople al servomotor

Cabe destacar que se ha tenido en cuenta la posición de colocación del eje correspondiente al engranaje movido por el servomotor, de manera que pueda accionar el mecanismo de 4 barras correspondiente a uno de los dos lados simétricos, como se ha explicado en apartados anteriores [Figura 4.37]:

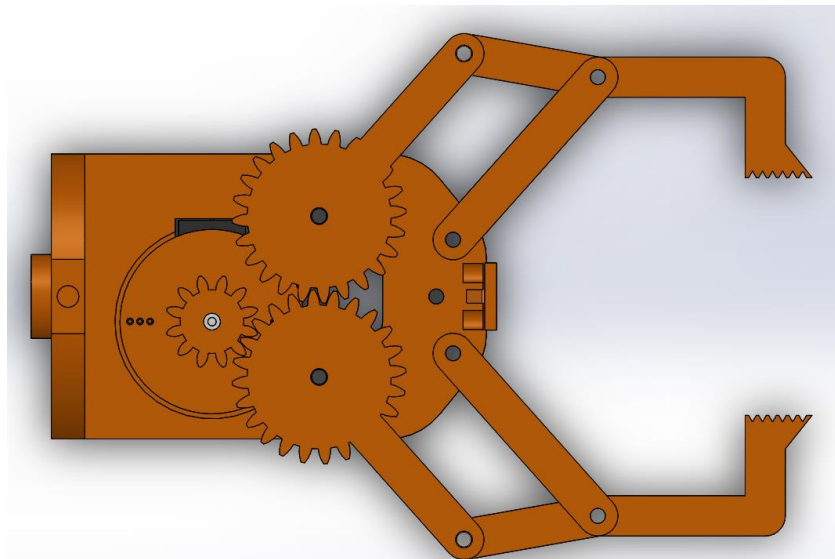


Figura 4. 37. Detalle colocación de los engranajes sobre la base

Se ha añadido un sistema de acoplamiento en el frontal de la herramienta que nos permitirá acoplar correctamente la herramienta ventosa [Figura 4.38]. Para ello se ha tenido en cuenta la geometría del acoplamiento dado por esta herramienta.

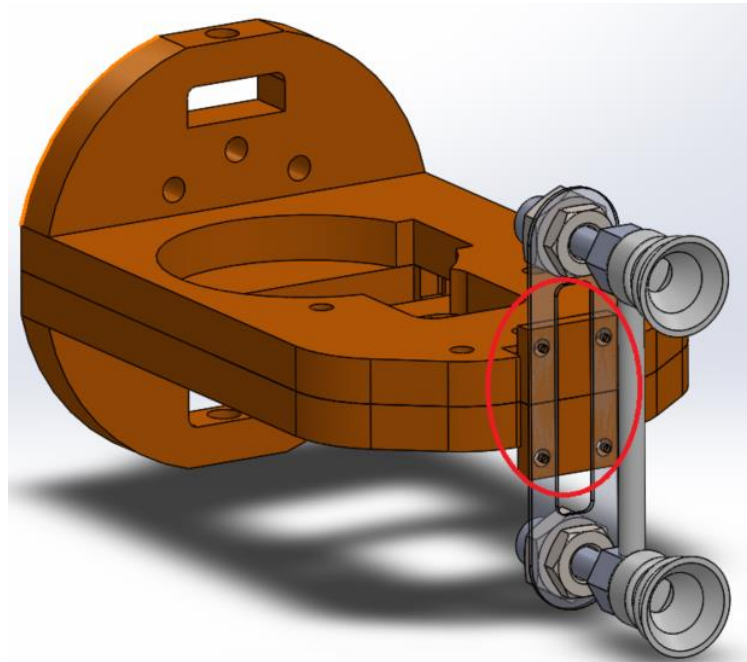


Figura 4. 38. Detalle acoplamiento herramienta de ventosas

Por último, se ha incluido el sistema de acople al robot ABB [Figura 4.39]. Dentro de este sistema también se incluye un taladro y una apertura que harán posible el acople al soporte tanto superior como inferior.

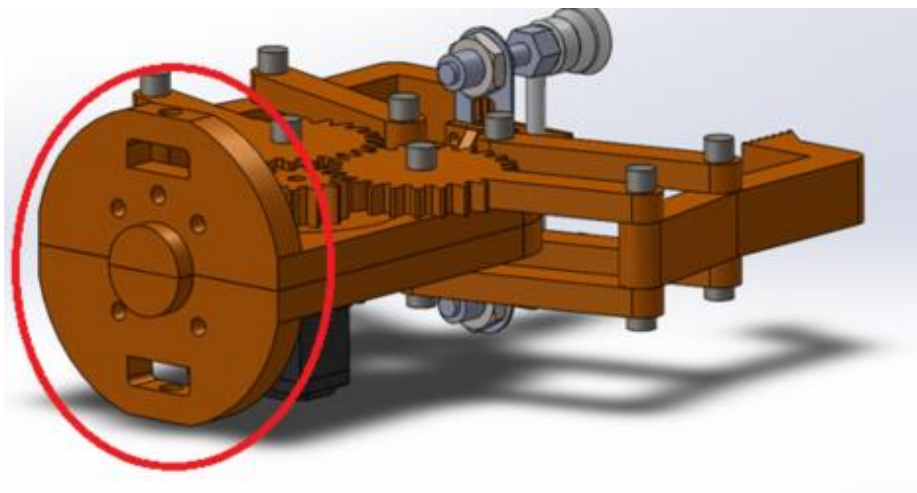


Figura 4. 39. Detalle acoplamiento al robot IRB140

4. SOPORTES PARA ELEMENTOS AUXILIARES

Dentro de nuestro diseño debemos tener en cuenta que se han de alojar todos los componentes necesarios para el funcionamiento de la herramienta. Mediante la inclusión de dos soportes que irán colocados en la parte superior e inferior de la base se conseguirán acoplar a la herramienta los siguientes sistemas:

- Arduino Uno para el control del servomotor y la ventosa.
- Arduino nano + módulo NRF24L01 para la recepción de las ordenes de actuación.
- Arduino Uno para el funcionamiento de la pantalla y el sensor de distancia.
- Pantalla LCD para mostrar información del sensor de distancia.
- Bomba y válvula del sistema de ventosas.

Soporte superior [Figura 4.40]:

En él irán acoplados:

- Arduino Uno para el control del servo y la ventosa.
- Arduino nano + módulo NRF24L01 para la recepción de las ordenes de actuación.

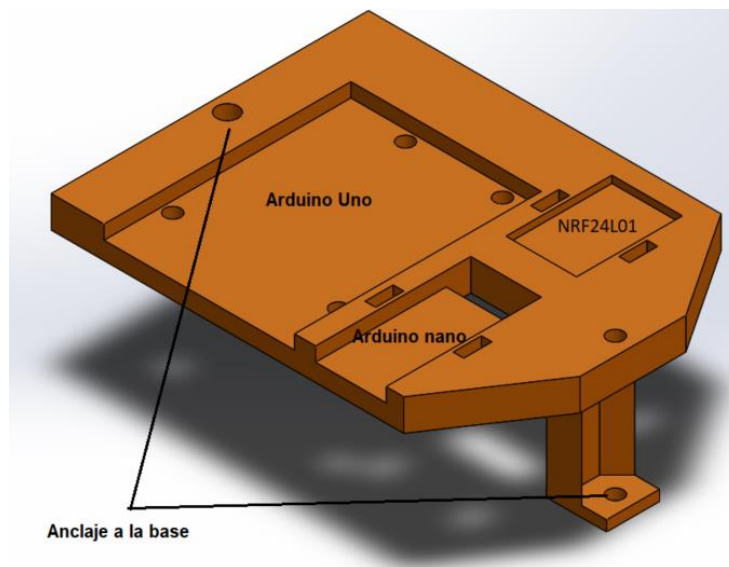


Figura 4. 40. Diseño del soporte superior

Esta base se colocará a 33 mm de altura con respecto a la base de la herramienta [Figura 4.41].

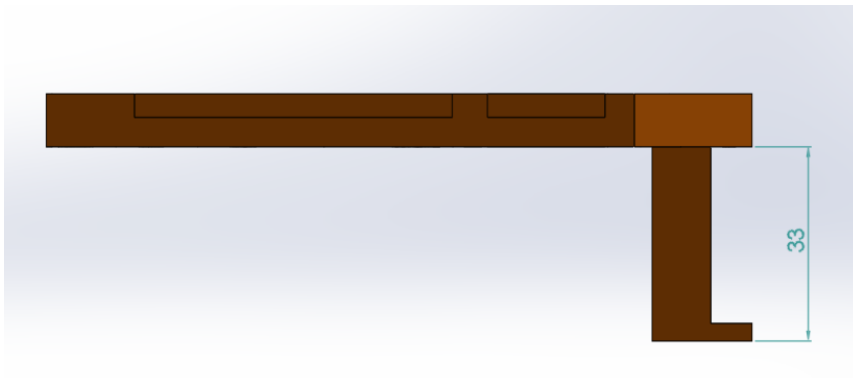


Figura 4. 41, Detalle de altura del soporte superior

Una vez se le acoplen los sistemas correspondientes el aspecto del soporte será el que se muestra en la figura 4.42 a continuación.

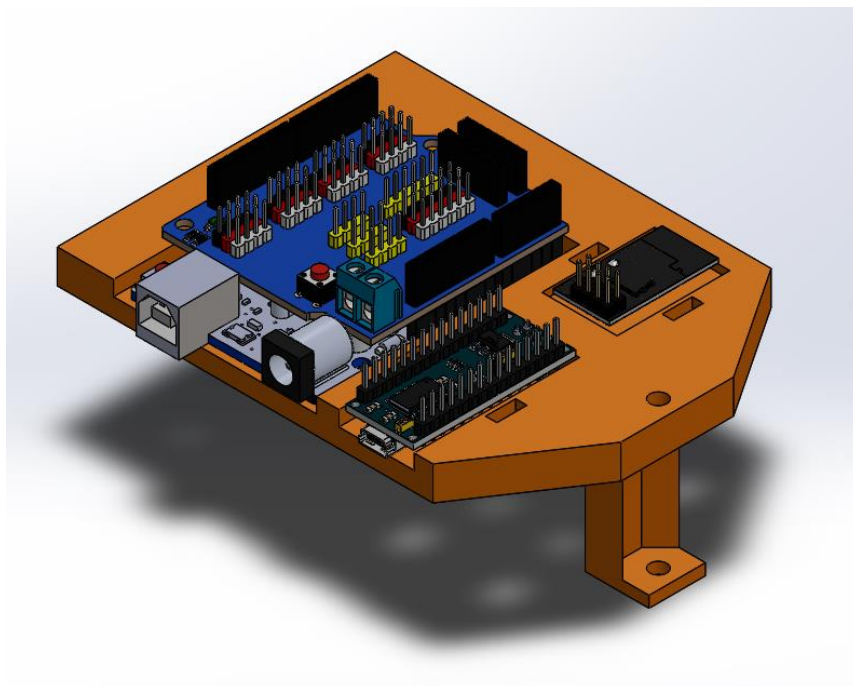


Figura 4. 42. Soporte superior junto con los elementos a acoplar en el

Soporte inferior [Figura 4.43]:

En él irán acoplados:

- Arduino Uno para el funcionamiento de la pantalla y el sensor de distancia.
- Pantalla LCD para mostrar información del sensor de distancia.
- Bomba y válvula del sistema de ventosas

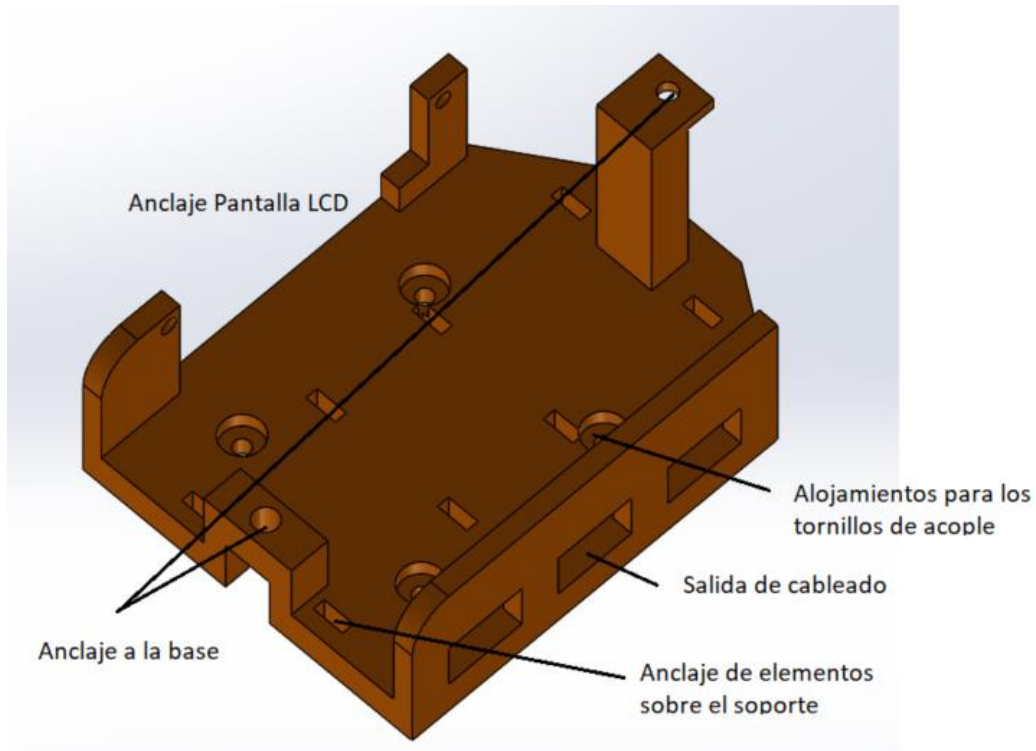


Figura 4. 43. Diseño soporte inferior

A este soporte se le ha incrementado la distancia a la base de la herramienta puesto que en este espacio ha de alojarse el servomotor del sistema de pinza, así como el Arduino de control de la pantalla [Figura 4.44].

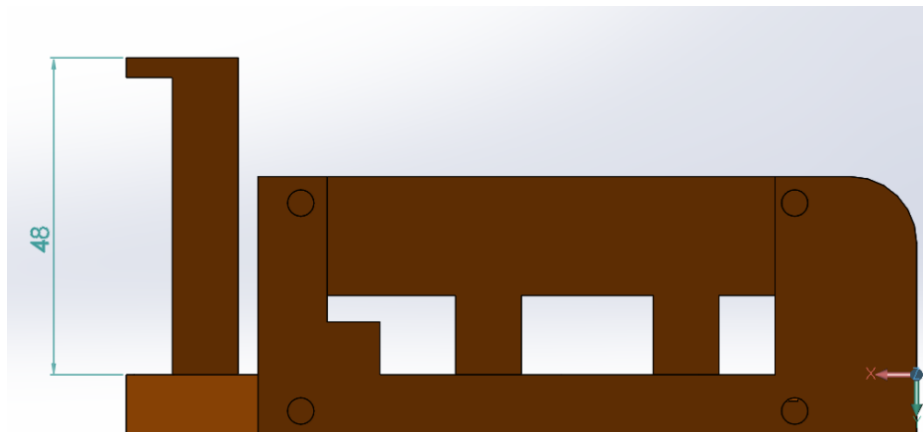


Figura 4. 44. Detalle altura soporte inferior

Una vez se le acoplen los sistemas correspondientes este será el aspecto del soporte [Figura 4.45]:

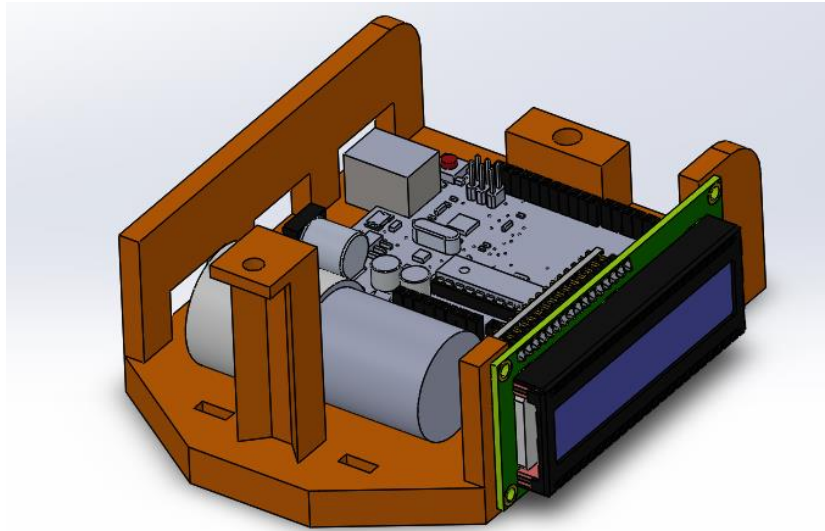


Figura 4. 45. Soporte inferior junto con los elementos a acoplar en el

5. CARCASA PARA SISTEMA DE ALIMENTACIÓN (“Caja Batería”)

Se incluye una carcasa que irá anclada al soporte inferior [Figura 4.46] y cuya finalidad será la de albergar la powerBank con la que se alimentará el sistema.

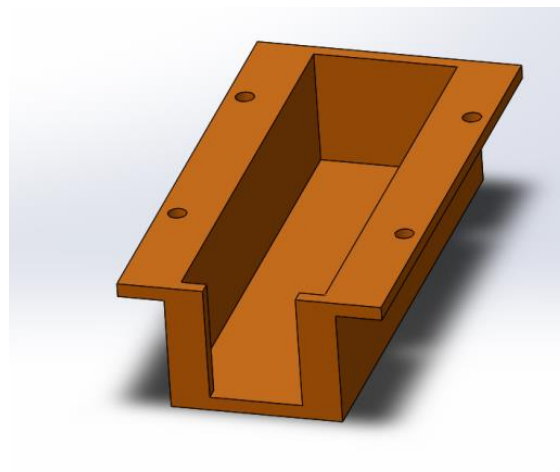


Figura 4. 46. Carcasa para sistema de alimentación

6. SOPORTE PARA EL SENSOR DE DISTANCIA

El último elemento que nos queda por incluir dentro de nuestra herramienta es el sensor de distancia.

Para ello se ha diseñado un acoplamiento que se situará al frente de la herramienta, quedando por detrás de la ventosa y en el plano medio de la herramienta. De esta manera no tendrá ninguna interferencia en la medida salvo cuando la pinza cierre por completo.

El acoplamiento es como se puede observar en la siguiente imagen [Figura 4.47]:

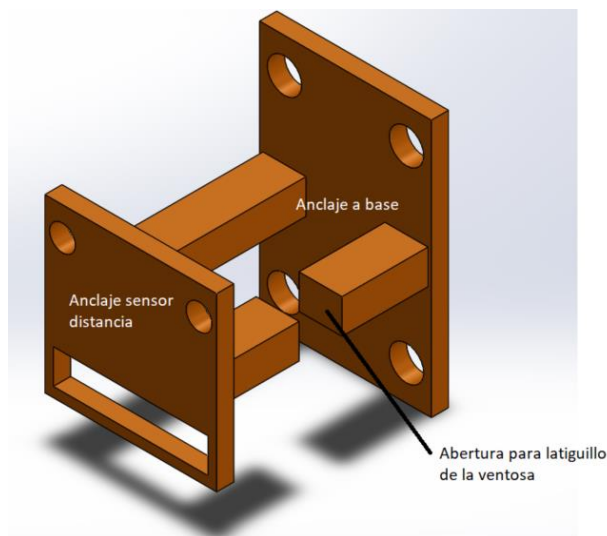


Figura 4. 47. Diseño soporte para sensor de distancia

Su colocación junto con la base y las ventosas es la siguiente [Figura 4.48]:

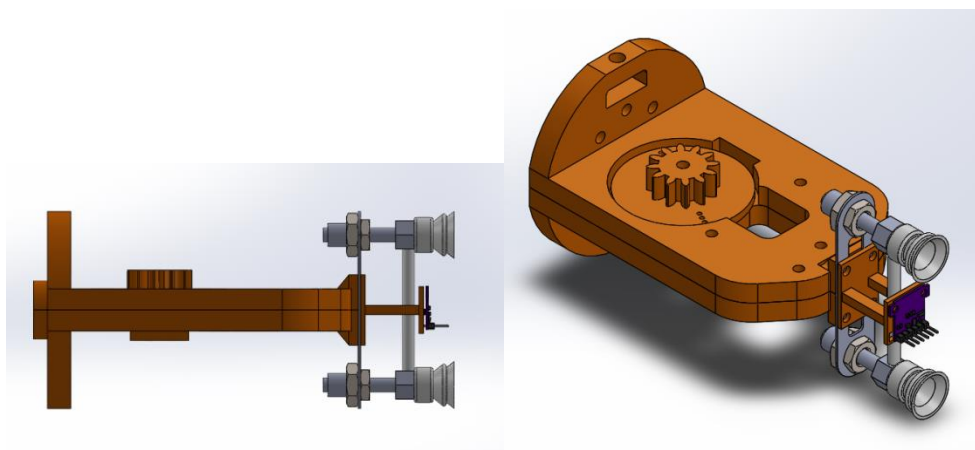


Figura 4. 48. Detalle de colocación para el soporte del sensor de distancia

Se le proporciona la distancia suficiente con respecto a la base para sobrepasar a la conexión entre ambas ventosas y que esto no interfiera en la medida.

4.1.3. DISTRIBUCION Y ENSAMBLE DEL CONJUNTO

Una vez diseñada cada una de las piezas que compondrá nuestra herramienta, tan solo quedará realizar el ensamblaje completo del sistema y verificar que no existen interferencias entre las distintas piezas o con los sistemas a incluir.

Como se ha visto en el apartado anterior, cada una de las piezas se ha diseñado teniendo en cuenta los distintos elementos, actuadores y sistemas que han de acoplar en ella.

De manera resumida podremos realizar una lista de las distintas piezas que van a componer el ensamblaje y de los actuadores o sistemas que lo harán funcionar [Tabla 12]:

Piezas diseñadas	Actuadores/sistemas
-Engranaje de acople al servomotor -Eslabones del sistema de pinza: <ul style="list-style-type: none"> • Eslabón simple • Eslabón con engranaje • Eslabón de pinza -Base de la herramienta: <ul style="list-style-type: none"> • Base superior • Base inferior -Soporte superior -Soporte inferior -Soporte para sistema de alimentación -Soporte para sensor de distancia	-Servomotor -Pieza de acoplamiento al servomotor -Sistema de ventosas -Arduino uno para control de pinza y ventosa -Sensor Shield v5.0 para Arduino uno -Arduino uno para pantalla y sensor de distancia -Arduino nano para control -Receptor de señal NRF24L01 -Pantalla LCD -Power Bank (sistema de alimentación)

Tabla 12. Relación piezas-actuadores

Se ha realizado también una tabla [Tabla 13] en la que se puede observar cada pieza diseñada junto con los elementos de los que es encargada de acoplar en ella:

PIEZA	ELEMENTOS QUE ACOPLAR
BASE (conjunto superior e inferior)	-Servomotor -Mecanismo 4 barras (pinza) -Sistema de ventosas -Acoplamiento para el sensor de distancia -Soporte superior -Soporte inferior
Soporte superior	-Arduino uno + sensor Shield para control de servomotor y sistema de ventosas. -Arduino nano -NRF24L01
Soporte inferior	-Pantalla LCD -Arduino uno para pantalla y sensor de distancia. -Bomba y válvula para las ventosas -Soporte para sistema de alimentación
Soporte para sistema de alimentación	PowerBank
Soporte para sensor de distancia	Sensor de distancia

Tabla 13. Pieza con cada uno de los elementos que deberá acoplar a ella

Este ensamblaje quedará perfectamente reflejado e indicado dentro del plano correspondiente. En él se podrán distinguir cada uno de sus elementos como se muestra a continuación en las figuras de la 4.49 a la 4.51.

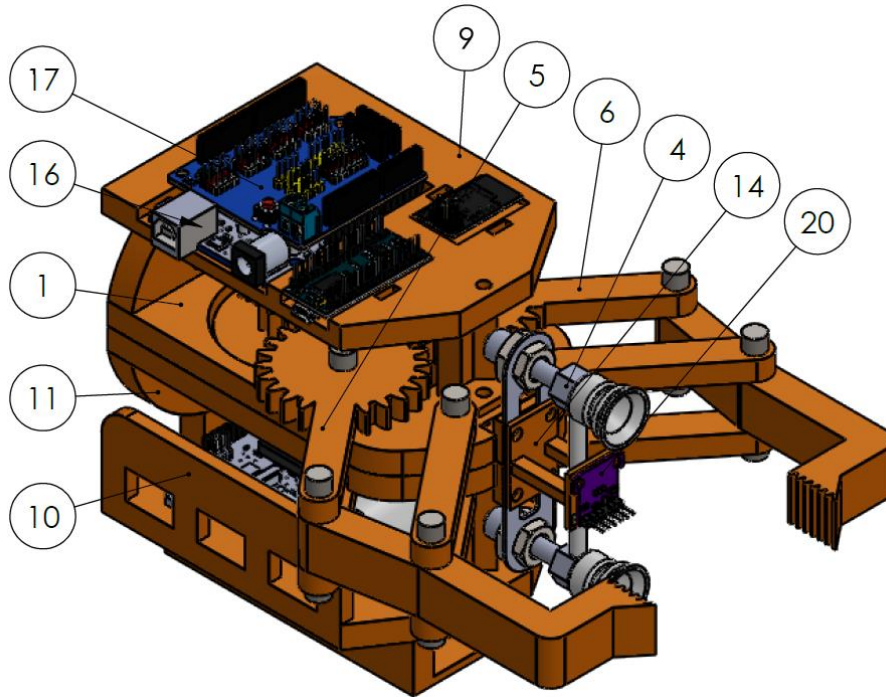


Figura 4. 49. Vista isométrica izquierda herramienta ensamblada

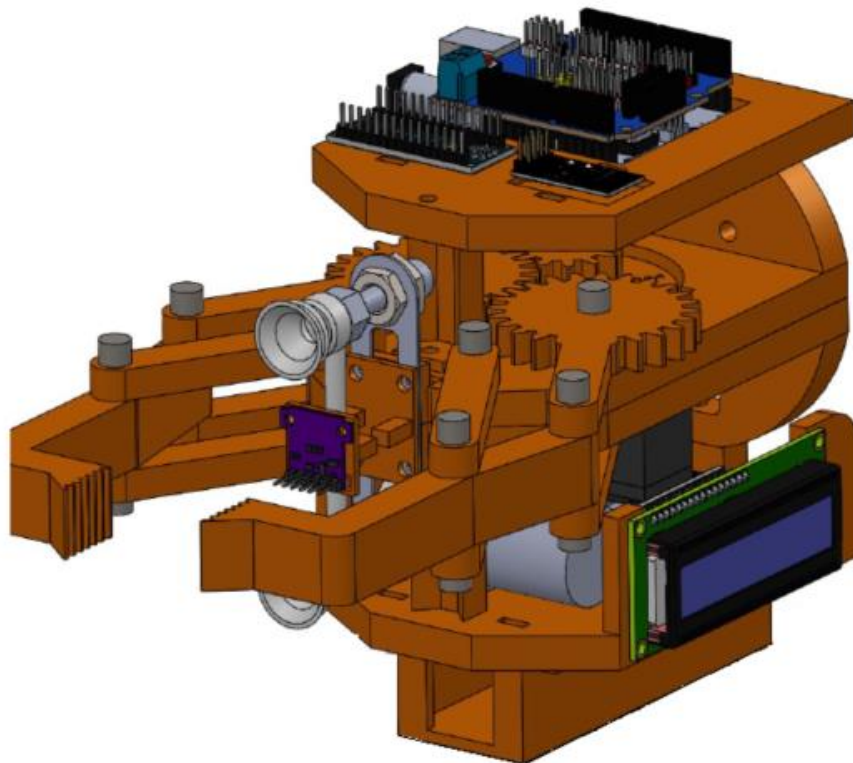


Figura 4. 50. Vista isométrica derecha herramienta ensamblada

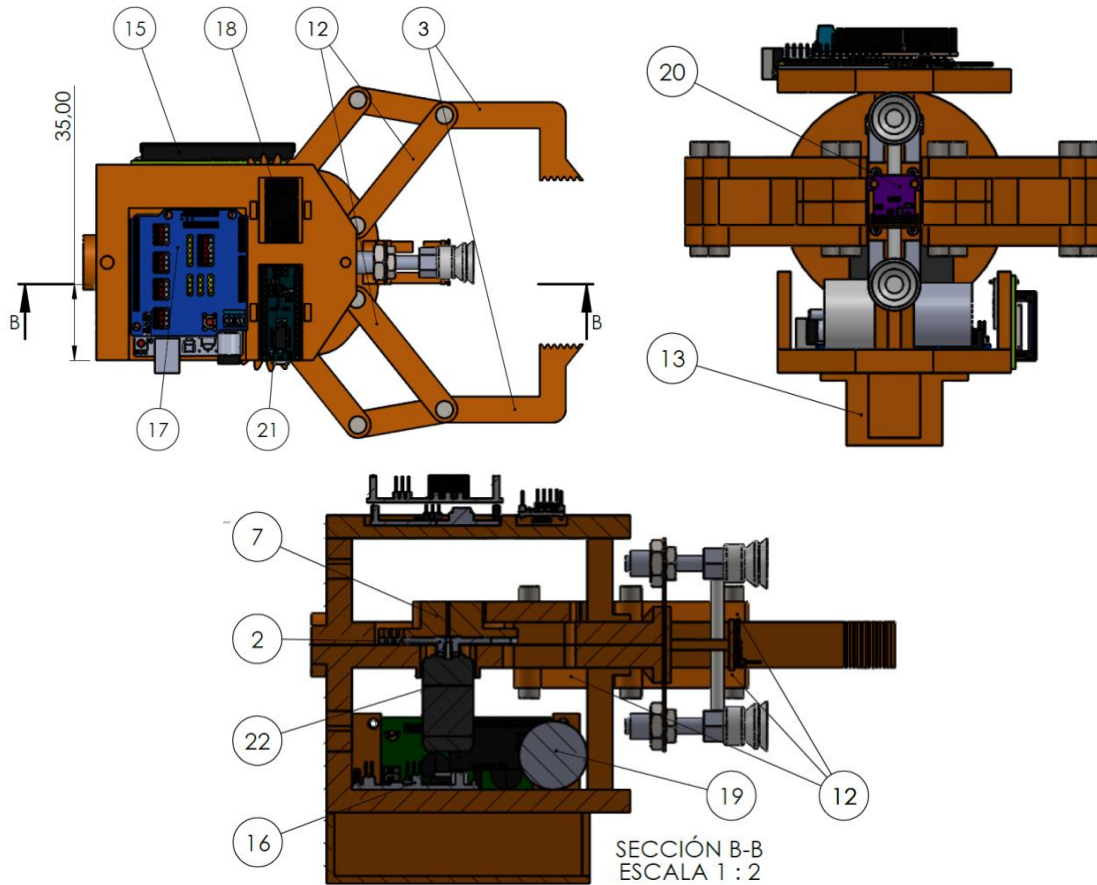


Figura 4. 51. Vistas ensamble herramienta

N.º DE ELEMENTO	N.º DE PIEZA	CANTIDAD			
1	BASE superior	1	12	Eslabon simple	6
2	Acople servo	1	13	CajaBateria	1
3	Eslabón pinza	2	14	Soporte sensor distancia	1
4	ENSAMBLAJE VENTOSAS	1	15	LCD AP214	1
5	Eslabon con engranaje modificado	1	16	Arduino Uno	2
6	Eslabón con engranaje	1	17	Arduino Sensor shield v5.0	1
7	Engranaje de acople al servo	1	18	nRF24L01	1
9	Soporte superior	1	19	Bomba_ventosa	1
10	Soporte inferior	1	20	Sensor distancia vl5310x	1
			21	arduino nano	1
			22	Servo Motor	1

Tabla 14. Piezas contenidas en el ensamble de la herramienta

4.1.3.4. COMPROBACION DE LAS INTERFERENCIAS ENTRE PIEZAS

Una vez ensamblados todos los elementos en la herramienta, se comprueba que no existe interferencia entre ninguna de las partes. Siendo el resultado totalmente satisfactorio.

Como comprobación adicional se muestran los puntos más problemáticos dentro del diseño:

- No existe interferencia entre ninguno de los elementos fijos que componen la herramienta.
- No existe interferencia entre el mecanismo de pinza y el sistema de ventosas situado en el frontal de la base al realizar el movimiento cierre del mecanismo [Figura 4.52].

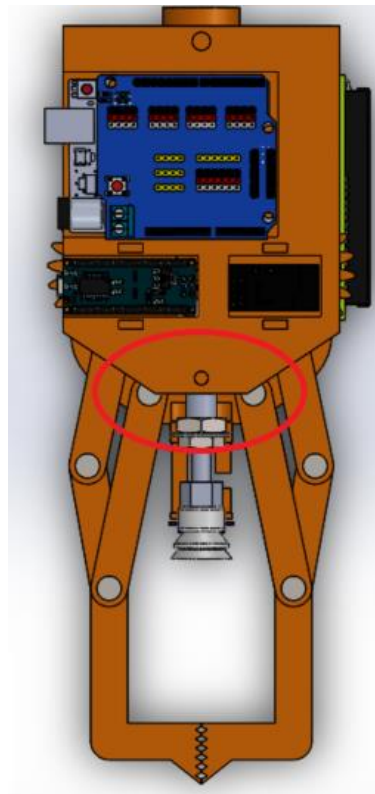


Figura 4. 52. Interferencia sistema de pinzas con elementos sobre la herramienta.

- No existe interferencia entre el mecanismo de pinza y ninguno de los dos soportes al realizar el movimiento de apertura y cierre del mecanismo [Figura 4.53 y 4.54].

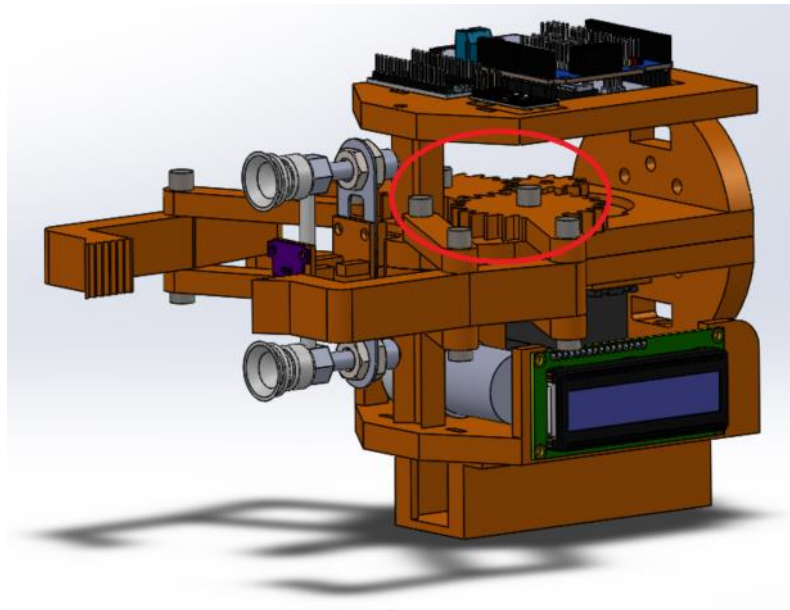


Figura 4. 53. Interferencia entre las pinzas y el soporte superior

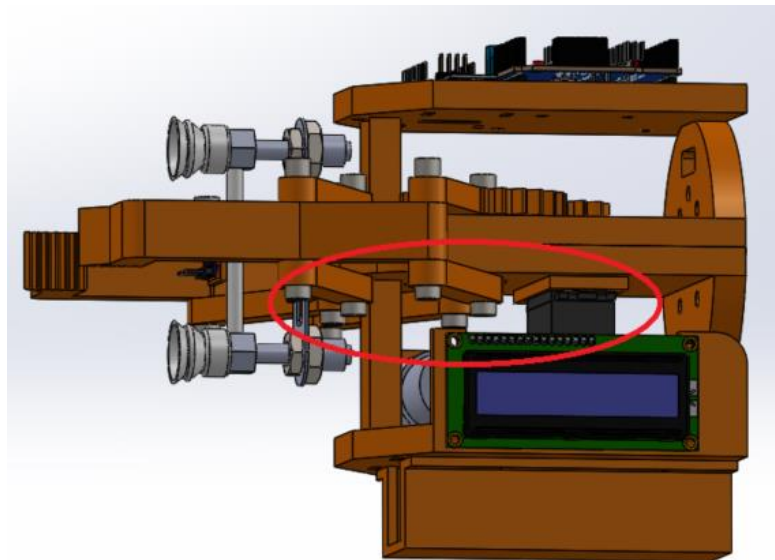


Figura 4. 54. Interferencia entre las pinzas y el soporte inferior

- No hay interferencia entre el mecanismo y el sensor de distancia. El único punto en el que el sensor de distancia se vea interferido por la proximidad de las pinzas será una vez que se esté realizando el cierre de las pinzas para manipular una pieza, lo cual se encuentra dentro de las especificaciones del diseño.

4.2. VERIFICACIÓN DE LAS PIEZAS MEDIANTE SOLIDWORKS

Una vez diseñadas todas las piezas de la herramienta se deberá realizar un estudio para comprobar que resistirán correctamente frente a los esfuerzos a los que estarán sometidas. Para ello se realiza un estudio mediante SolidWorks Simulation de los elementos sometidos a los esfuerzos más representativos.

Se realizará el estudio sobre 3 de las piezas que conforman el ensamblaje, pues serán las que mayor sollicitación de peso tengan. Los elementos que ha de soportar y peso de cada uno de ellos se muestran en la tabla 15:

PIEZA DE ESTUDIO	COMPONENTE QUE SOPORTA	PESO [g]	FUERZA [N]
Soporte superior	Arduino uno (1)	27	0,2646
	Arduino Shield v5.0	25	0,245
	Arduino nano (2) + Módulo de comunicación nRF24L01	13	0,1274
Soporte inferior	Bomba y válvula sistema ventosa	90	0,882
	Pantalla LCD	31	0,3038
	Arduino uno	27	0,2646
	PowerBank (sistema de alimentación)	65	0,637
Base	Peso correspondiente a los elementos soportados por la base superior	65	0,637
	Soporte superior	58	0,5684
	Soporte inferior	77	0,7546
	Peso soportado por la base inferior	213	2,0874
	Servomotor	43	0,4214
	Pinzas	98	0,9604
	Ventosas	74	0,7252
	Soporte sensor de distancia y sensor	7	0,0686

Tabla 15. Desarrollo de componentes que intervendrán dentro del estudio de SolidWorks Simulation

El primer paso que se ha realizado es la creación del material con el que se han fabricado estas piezas dentro del entorno de materiales de SolidWorks, puesto que la base de datos de materiales no se encuentra aún actualizada y no lo contiene.

El material de fabricación es el PLA y sus datos introducidos dentro de SolidWorks son los que se muestran en la siguiente figura 4.55.

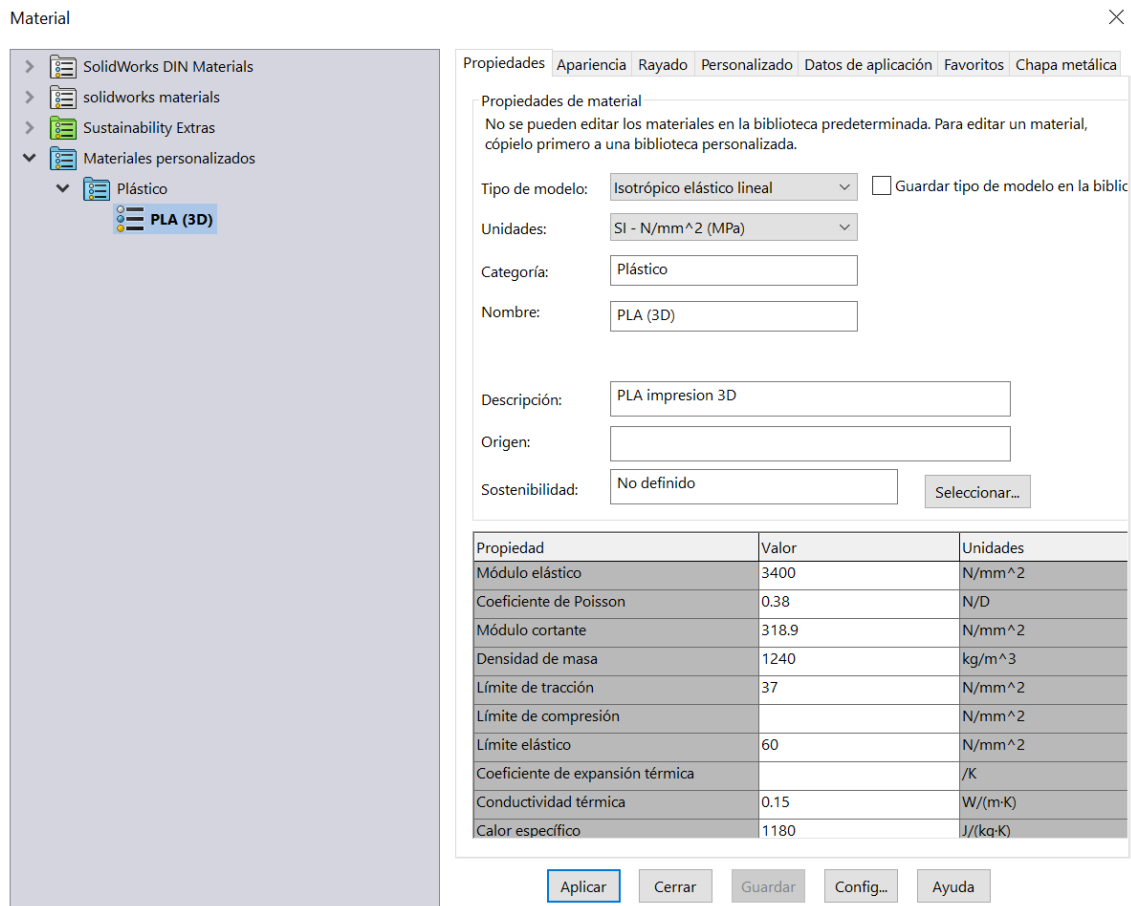


Figura 4. 55. PLA Solidworks

Se ha incluido también dentro de este material la apariencia real del filamento que se usará, puesto que este es de color naranja.

Una vez aplicado el material correcto a las distintas piezas se ha realizado el estudio de cada una de ellas. Este estudio se encuentra dentro del punto 5 del anexo de cálculos: “5. ANÁLISIS DE LAS PIEZAS MEDIANTE SOLIDWORKS SIMULATION”.

A continuación, se muestran los resultados obtenidos para cada uno de los componentes de estudio, junto con las consideraciones tomadas en función de estos resultados.

4.2.1. SOPORTE SUPERIOR

Resultados del análisis mediante SolidWorks [Tabla 16]:

Tensión máxima	4,5e+04 N/m² (< Límite elástico=6,000e+07 N/m²)
Desplazamiento máximo	1,959e-03 mm

Tabla 16. Resultados estudio soporte superior

En lo referente a las tensiones soportadas por la pieza, el resultado es satisfactorio pues el límite elástico no se supera en ningún momento y queda lejos del rango de tensiones alcanzado en el modelo.

Para los valores obtenidos de desplazamientos, el resultado se considerará satisfactorio puesto que estos valores son muy pequeños.

4.2.2. SOPORTE INFERIOR

Resultados del análisis mediante SolidWorks [Tabla 17]:

Tensión máxima	1,546e+05 N/m² (< Límite elástico=6,000e+07 N/m²)
Desplazamiento máximo	3,425e-03 mm

Tabla 17. Resultados estudio soporte inferior

En lo referente a las tensiones soportadas por la pieza, el resultado es satisfactorio pues el límite elástico no se supera en ningún momento y queda lejos del rango de tensiones alcanzado en el modelo.

Para los valores obtenidos de desplazamientos, el resultado se considerará satisfactorio puesto que estos valores son muy pequeños.

4.2.3. CONJUNTO BASE:

El conjunto formado por las dos mitades de la base y los elementos que forman la pinza se ha considerado el más desfavorable a la hora de determinar la carga máxima que puede manipular la herramienta, esto es debido a que es el elemento responsable de soportar un mayor número de componentes y esfuerzos.

Para su estudio se ha realizado un modelo en el que se simplifica la geometría básica de los componentes que soportaran los esfuerzos a los que está sometida la base. En este modelo se han simplificado los engranajes para una mayor eficiencia de cálculo. [Figura 4.56]

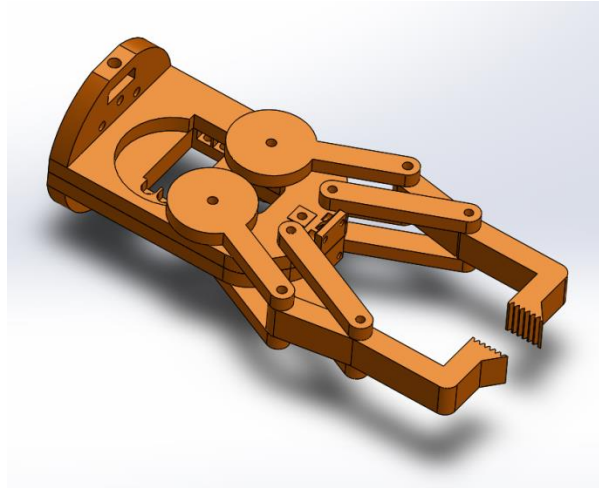


Figura 4. 56. Simplificación de la herramienta para simulación

La posición en la que se estudiarán los esfuerzos es la que se ha considerado más desfavorable, expuesta a continuación [Figura 4.57]:

- Pinza en posición horizontal: el esfuerzo generado por el momento de la carga que se maneja a través de las pinzas será mayor, ya que está más alejado de la base de la herramienta.
- Pinza en apertura lo menor posible: se ha considerado que se va a manipular una pieza de 25 mm de anchura, por lo que las pinzas se encontrarán separadas esa distancia entre sí.

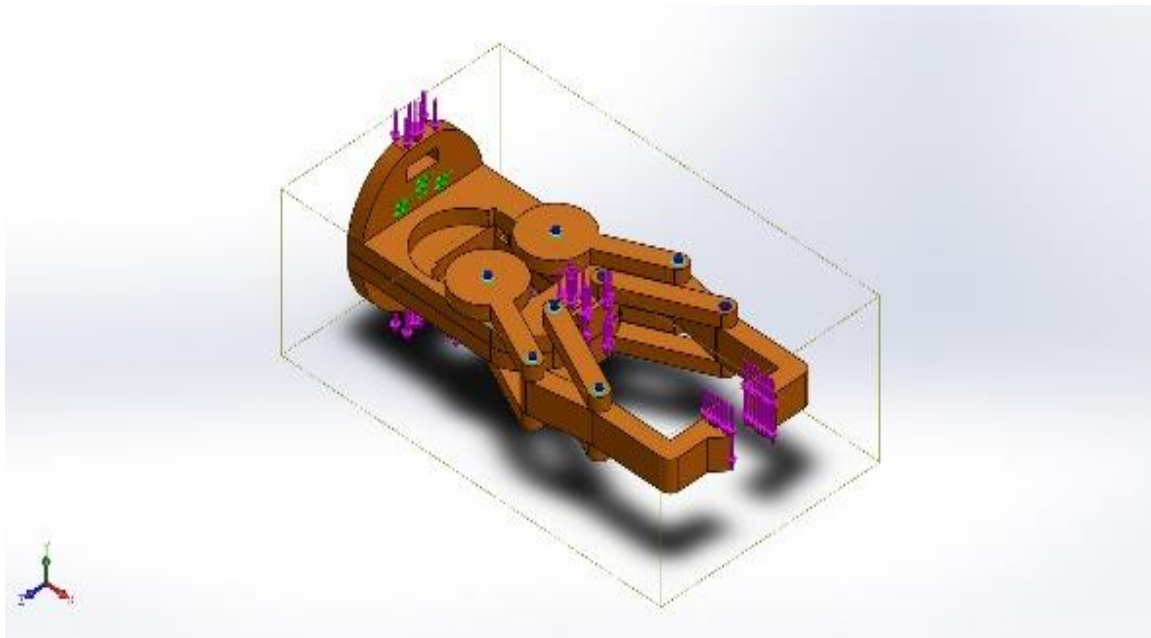


Figura 4. 57. Posición de estudio y fuerzas aplicadas

Todos los datos correspondientes a las fuerzas aplicadas y su posición pueden verse con detalle dentro del apartado correspondiente del anexo de cálculos.

Se estudiará la herramienta con 3 valores distintos de esta fuerza para determinar cuál ha de ser la carga máxima de utilización de esta:

- **Estudio 1:** 1kg; aproximación a 10N
- **Estudio 2:** 2kg; aproximación a 20N
- **Estudio 3:** 0,5kg; aproximación a 5N. Tras realizar el estudio con un gran rango de pesos, el tercer estudio que se mostrará dentro de este proyecto corresponde al límite de peso que se ha considerado como máximo para la utilización de la herramienta.

Los valores que se han obtenido en cada uno de estos estudios son los siguientes:

RESULTADOS DEL ESTUDIO PARA 1 KG [Tabla 18]

Tensión máxima	2,739e+07 N/m² (< Límite elástico=6,000e+07 N/m²)
Desplazamiento máximo	7,82 mm

Tabla 18. Resultados estudio base 1kg

El resultado correspondiente al valor de tensión alcanzado es satisfactorio pues el límite elástico no se supera en ningún momento. Sin embargo, los valores de desplazamiento de este estudio alcanzan los 7,82 mm. Se considerará un valor alto pues se encuentra demasiado próximo al desplazamiento de 1cm que se ha considerado como excesivo para esta herramienta.

RESULTADOS DEL ESTUDIO PARA 2 KG [Tabla 19]

Tensión máxima	5,436e+07 N/m² (< Límite elástico=6,000e+07 N/m²)
Desplazamiento máximo	15,5 mm

Tabla 19. Resultados estudio base 2kg

El resultado de las tensiones existentes, aun quedando por debajo del límite elástico, se acerca demasiado a él. Es por ello por lo que consideraremos 2 kg como una carga la cual nuestro sistema no deberá manipular. No se estudiarán fuerzas mayores de 2kg ya que esto nos daría como resultado sobrepasar el límite elástico de nuestro material.

De igual manera, los valores de desplazamiento de este estudio alcanzan los 15,5 mm y se considerarán un valor demasiado alto para la deformación de la herramienta.

RESULTADOS DEL ESTUDIO PARA 0,5 KG [Tabla 20]

Tensión máxima	1,386e+07N/m ² (< Límite elástico=6,000e+07 N/m ²)
Desplazamiento máximo	3,95 mm

Tabla 20.Resultados estudio base 0,5kg

El resultado correspondiente al valor de tensión alcanzado es satisfactorio pues el límite elástico no se supera en ningún momento.

Los valores de desplazamiento de este estudio alcanzan los 3,95 mm. Se considerará este valor como el último aceptable ya que no se quiere que el modelo pase de los 5 mm contando con un criterio de seguridad de mayorar estos valores al 110%.

Valor de deformación máximo permitido = 3,95 · 1,1 = 4,34mm < 5mm → Aceptable

4.2.4. RESULTADOS DEL ESTUDIO MEDIANTE SOLIDWORKS:

Tras el estudio realizado mediante simulación en SolidWorks, podemos afirmar que el modelo diseñado cumple con los requisitos satisfactoriamente. Todos sus elementos superan las pruebas realizadas y serán capaces de funcionar de manera correcta una vez ensamblados.

Se ha establecido una carga máxima de utilización de 500g para las cargas a manipular por la herramienta.

4.3. FABRICACIÓN MEDIANTE IMPRESIÓN 3D

Una vez superada la fase de diseño, el siguiente paso es la fabricación de cada una de las piezas que componen el sistema. Para ello se va a utilizar la técnica de impresión 3D en PLA mediante la impresora que se encuentra en las instalaciones de la universidad de Almería.

En este apartado se muestra el proceso resultante de llevar el diseño creado digitalmente mediante SolidWorks hasta su obtención física. [4] Esto ha conllevado una serie de pasos intermedios que se muestran a continuación.

4.3.1. TRANSFERENCIA DE ARCHIVOS

Todas las piezas que se han diseñado han sido realizadas en el formato propio de SolidWorks (SLDPRT para piezas y SLDASM para ensamblajes). Estos formatos son formatos registrados propiedad de SolidWorks.

Para poder transferir estos diseños desde el formato registrado de SolidWorks al programa Ultimaker Cura con el que generaremos el código para su posterior impresión en 3D, se han de transferir los archivos desde su formato de origen a el formato STL.

El formato STL se trata de un formato neutro que nos permitirá transferir las geometrías de un programa al otro. Este formato genera la geometría del sólido a partir un conjunto de vértices, aristas y triángulos interconectados, donde como mínimo dos triángulos comparten cada arista y cada vértice. Otros elementos como puntos, líneas o colores se omiten.

Se muestra a continuación el modelo correspondiente a la base superior tras su conversión al formato STL [Figura 4.58]:

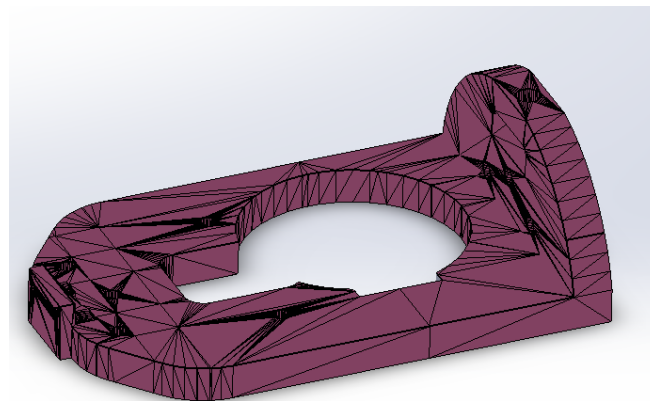


Figura 4. 58. Mallado de pieza en formato STL

4.3.2. PARÁMETROS CARACTERÍSTICOS

Una vez se han convertido los archivos al formato STL se procederá a el uso de Ultimaker CURA, el software por el cual podremos obtener los archivos de código g a introducir en la impresora 3D.

La función de Ultimaker CURA es la de, partiendo del archivo STL, generar las trayectorias que debe seguir el extrusor de la impresora para crear el modelo físico. Se generan las sucesivas capas de impresión en el plano XY, las cuales superponiéndose las unas con las otras en el eje Z se consigue un objeto en 3 dimensiones.

Al cargar el modelo, CURA generará automáticamente el código GCODE, ofreciendo una estimación del tiempo de impresión y gasto de material

El primer paso para obtener nuestro código g que nos permita la correcta impresión de las distintas piezas es el de configurar todos los parámetros dentro del software de CURA.

Como ejemplo, se muestra el proceso de obtención de la pieza correspondiente a la base de la herramienta a través de este software.

Los pasos para seguir son:

- 1. Configuración de la impresora dentro de CURA.** [Figura 4.59] Una vez instalado el software, se ha de especificar el modelo de impresora con el que se va a trabajar. En nuestro caso se trata del modelo WitBox 2 de la marca BQ.

Una vez se indica el modelo de impresora, el programa incluirá los parámetros correspondientes a la misma, como lo son el tamaño de la cama de impresión, la existencia o no de una cama calefactada para la impresión o la posibilidad de incluir distintos extrusores.



Figura 4. 59. Parámetros de la impresora dentro de UltimakerCura

- Apertura y colocación de las piezas.** [Figura 4.60] Se cargará la pieza que se desee fabricar y se seleccionará la posición en la que se desea obtener dentro de la base de impresión.

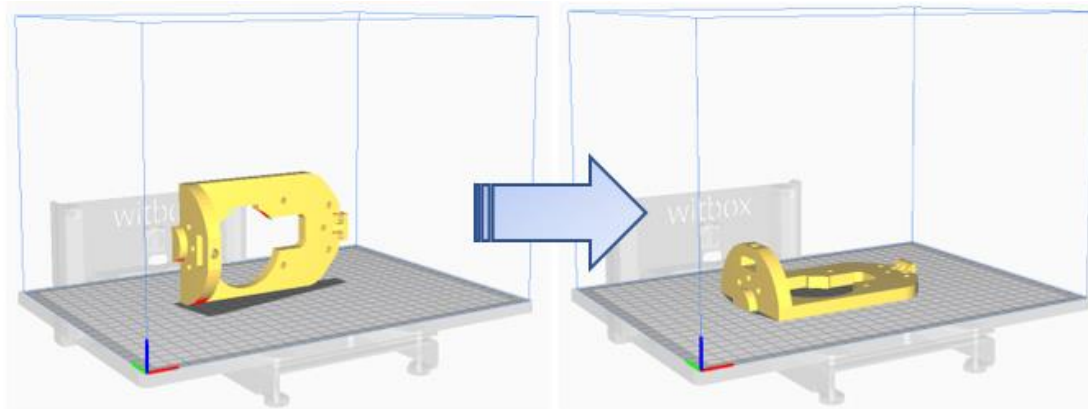


Figura 4. 60. Posicionamiento de las piezas en el entorno UltimakerCura

3. Selección de los parámetros de impresión dentro de nuestro software:

En este caso, para seleccionar los parámetros óptimos se ha partido de los recomendados para el modelo de impresora y se han modificado los siguientes:

- **Altura de capa:** constituye el avance en el eje z, es decir, cada capa que genera la máquina deposita el espesor aquí indicado. Por término general, el valor de este parámetro suele oscilar entre 0,05 mm y el diámetro de la boquilla de nuestro extrusor, que en este caso es de 0,4 mm.

Se trata de uno de los parámetros que más peso tienen a la hora del acabado, ya que una altura de capa muy baja generará sólidos con muy buenos acabados, pero aumentará el tiempo de impresión.

Para todas las piezas de esta herramienta se va a configurar una altura de capa de 0,18 mm, la cual deja un buen acabado, una buena adhesión entre capas y en un tiempo de impresión razonable.

- **Grosor de pared:** Espesor existente en el perímetro del sólido. CURA genera la trayectoria para que las superficies del modelo siempre se vean sólidas. Este parámetro es un múltiplo entero del diámetro de la boquilla del extrusor.

Para todas las piezas de esta herramienta se va a configurar un grosor de pared de 1,2 mm, que nos proporcionará una adecuada resistencia.

Este parámetro generará automáticamente el número de líneas de pared necesarias para conseguir el grosor deseado.

- **Patrón y densidad de relleno:** para optimizar la cantidad de material utilizado y disminuir los tiempos de impresión, en el interior de las superficies del modelo se genera un patrón formado por líneas que se entrecruzan. De este modo, no será necesario rellenar de material todo el interior de la pieza. Hay distintos tipos de patrones: triángulos, panal de abeja, cuadrados, concéntricos, etc.

Estos patrones aportan solidez a la pieza y su tamaño depende de la densidad que se introduzca. La densidad se puede variar desde el 0% (volumen vacío) hasta el 100% (pieza maciza).

Para todas las piezas de esta herramienta se va a configurar un patrón triangular por ser el que mejores características mecánicas ofrece. El parámetro de relleno se ha establecido en un 20% para las piezas que formarán nuestro diseño ya que sus caras se encuentran muy próximas y esto ya le proporciona la rigidez adecuada.

- **Temperatura:** dependerá del material que se esté imprimiendo y el fabricante de dicho material. Para el PLA esta temperatura suele oscilar entre 190 y 210°C.

Para todas las piezas de esta herramienta la temperatura de impresión se ha fijado en 200°C, tal como recomienda el fabricante.

- **Velocidades:** velocidad de impresión de cada una de las zonas de impresión medida en mm/s. El relleno puede realizarse a una mayor velocidad al no necesitar un buen acabado. En cambio, el perímetro exterior exige menores velocidades que el resultado sea bueno visualmente. Cada impresora tiene un rango propio de velocidades. En este caso se ha impuesto una velocidad de impresión de 60 mm/s para las caras exteriores debido a las recomendaciones de la impresora y la experiencia previa.
- Se han incluido los **soportes** para la correcta obtención de las geometrías sin apoyo directo.

Una vez seleccionados, se podrán observar los parámetros seleccionados directamente sobre el menú correspondiente dentro del software [Figuras 4.61 a 4.63]:

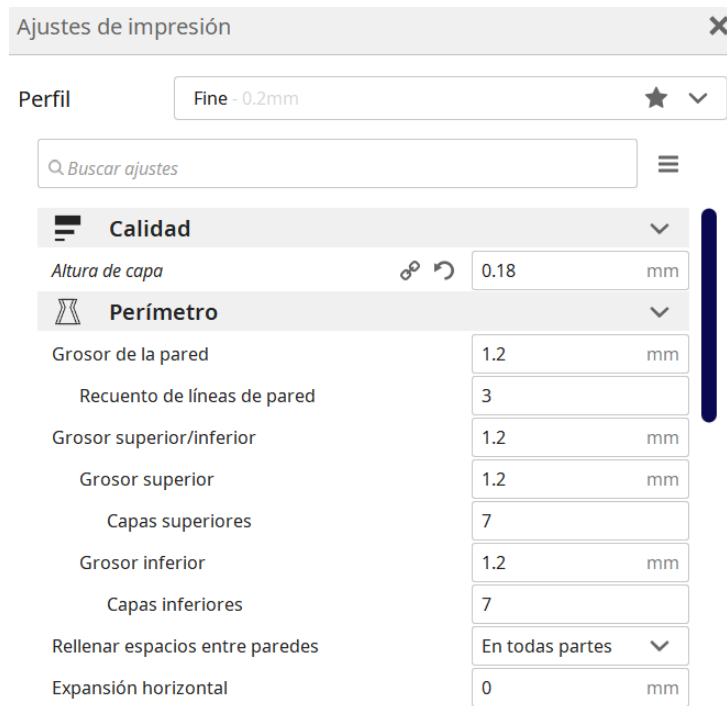


Figura 4. 61. Parámetros de impresión 3D (1)

	Relleno	
Densidad de relleno	20	%
Patrón de relleno	Rejilla	▼
Superposición del relleno	0.04	mm
	Material	
Temperatura de impresión	200	°C
Flujo de capa inicial	100	%
	Velocidad	
Velocidad de impresión	60	mm/s
	Desplazamiento	
Habilitar la retracción	<input checked="" type="checkbox"/>	
Salto en Z en la retracción	<input type="checkbox"/>	

Figura 4. 62. Parámetros de impresión 3D (2)

	Refrigeración	
Activar refrigeración de impresión	<input checked="" type="checkbox"/>	
Velocidad del ventilador	100.0	%
	Soporte	
Generar soporte	<input checked="" type="checkbox"/>	
Colocación del soporte	En todos sitios	▼
Ángulo de voladizo del soporte	50	°
	Adherencia de la placa de impresión	
Tipo adherencia de la placa de impresión	Borde	▼
	Extrusión doble	

Figura 4. 63. Parámetros de impresión 3D (3)

4.3.3. PIEZAS RESULTADO

Una vez se han configurado correctamente todos los parámetros de impresión, podemos realizar la segmentación de la pieza, por la cual el programa nos generará el código g con el que va a trabajar la impresora. Este código se ha de guardar en la unidad extraíble correspondiente que posteriormente se insertará en la impresora para su lectura [Figura 4.64].

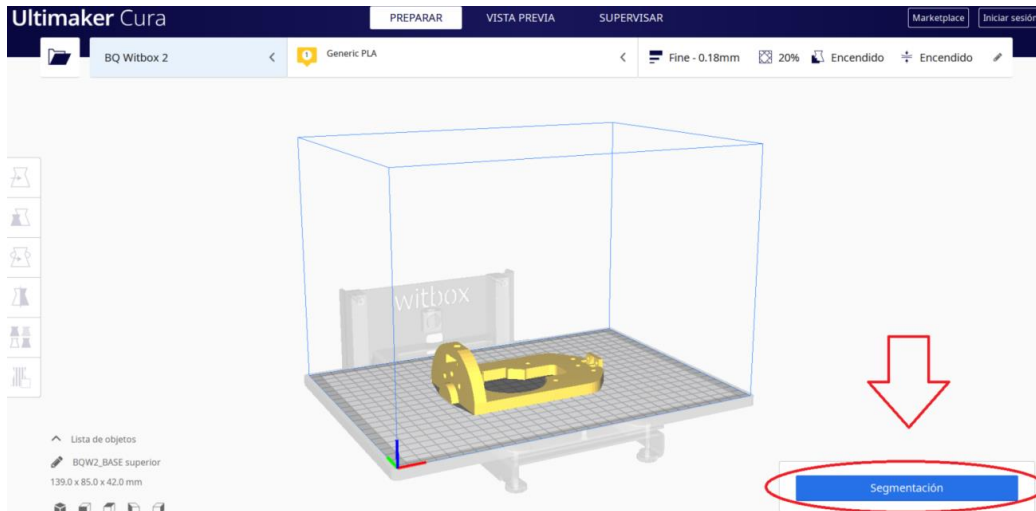


Figura 4. 64. Entorno de impresión final

Una vez se realiza la segmentación obtenemos el tiempo de impresión aproximado y el peso se la pieza [Figura 4.65]:

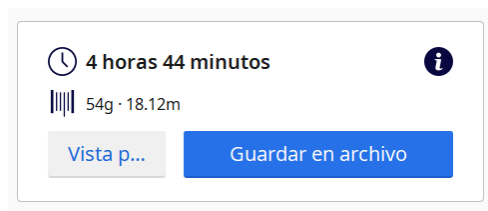


Figura 4. 65. Detalle de datos aportados por Cura

Se puede visualizar también la vista previa de la pieza, donde se comprueba que no exista ningún defecto [Figura 4.66]:

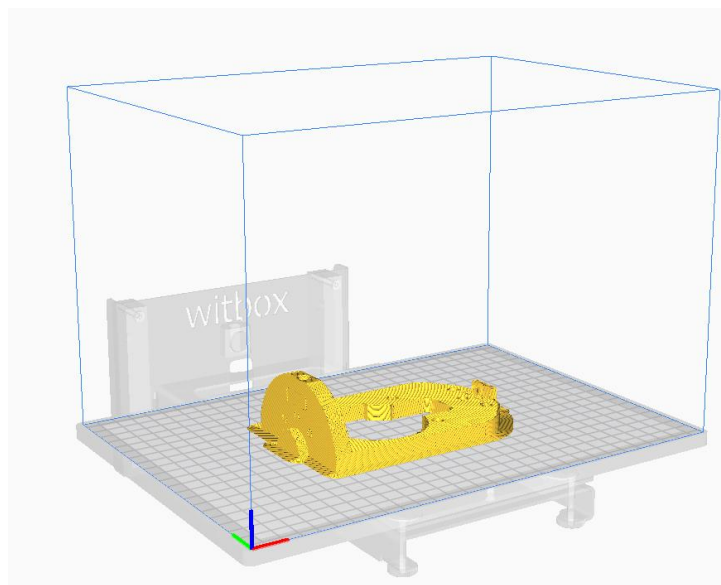


Figura 4. 66. Vista previa de la pieza a imprimir

Este proceso se realizará para cada una de las piezas que componen la herramienta. En la siguiente tabla 21 se puede ver las distintas piezas impresas junto con el tiempo de impresión y el peso correspondiente:

IMPRESIÓN 3D					
	Pieza	t impresión	t [min]	Unidades	peso
1	BASE superior	4h 44 min	284	1	54g
2	BASE inferior	5h 38min	338	1	64g
3	Eslabón simple	30 min	30	6	6g
4	Eslabón pinza	1h 28 min	88	2	17g
5	Acople servo	1h 6min	66	1	12g
6	Eslabón con engranaje 1	1h 10 min	70	1	14g
7	Eslabón con engranaje 2	1h 10 min	70	1	14g
8	Soporte superior	5h 13min	313	1	58g
9	Soporte Inferior	7h 7min	427	1	77g
10	Caja batería	4h 37min	277	1	52g
11	Soporte Sensor distancia	30min	30	1	5g
TIEMPO TOTAL		33h 13min	1993		

Tabla 21. Piezas, tiempos y pesos de impresión 3D

Los resultados correspondientes a las distintas piezas impresas son los siguientes [Figuras desde 4.67 a 4.74]:

Figura 4. 67. BASE SUPERIOR



Figura 4. 68. BASE INFERIOR

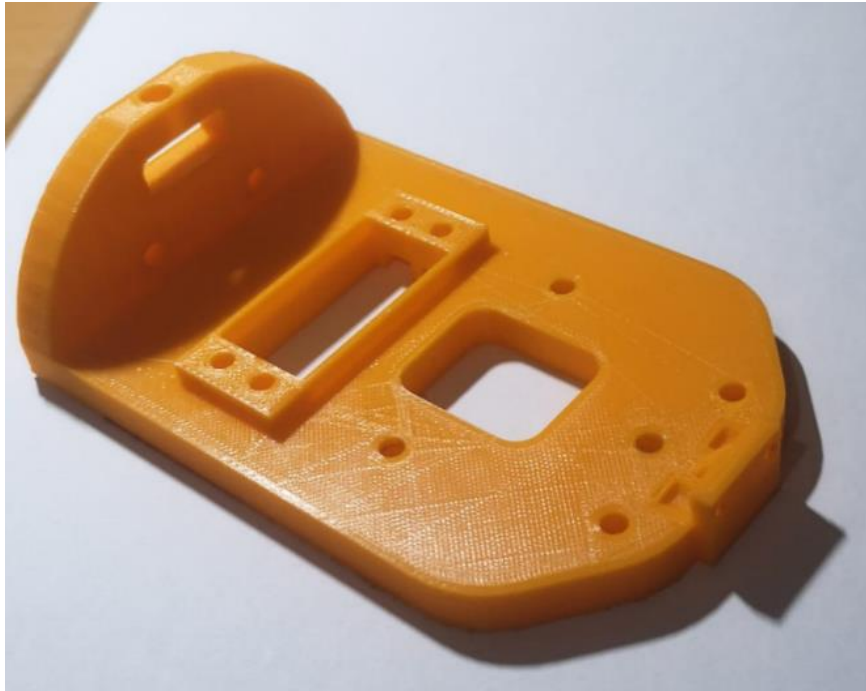


Figura 4. 69.ESLABONES SIMPLES (6), ESLABONES PINZA (2) Y ESLABONES CON ENGRANAJE (1 Y 2)



Figura 4. 70. ACOPLAMIENTO AL SERVOMOTOR

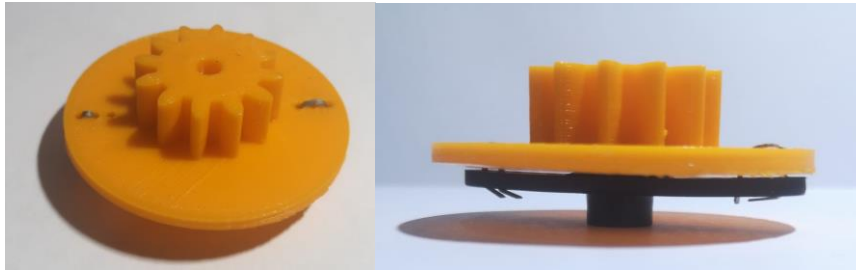


Figura 4. 71. SOPORTE SUPERIOR



Figura 4. 72. SOPORTE INFERIOR



Figura 4. 73.CAJA BATERIA



Figura 4. 74. SOPORTE PARA SENSOR DE DISTANCIA



4.4. HERRAMIENTA FINAL COMPLETA

Con tal de verificar la correcta fabricación e interconexión de los elementos impresos en 3D con los elementos a acoplar para hacer funcionar el sistema y entre ellos, se estudia en este punto el ensamblaje final de la herramienta y los subensamblajes realizados para llegar hasta él.

Los ensamblajes a estudiar se mostrarán según el orden seguido para el montaje del sistema completo robot-herramienta en las instalaciones de la Universidad de Almería.

1. Ensamblaje del soporte superior [Figura 4.75]. Los elementos que ha acoplar en ella serán:

- Arduino uno + sensor Shield v5.0.
- Arduino nano
- NRF24L01

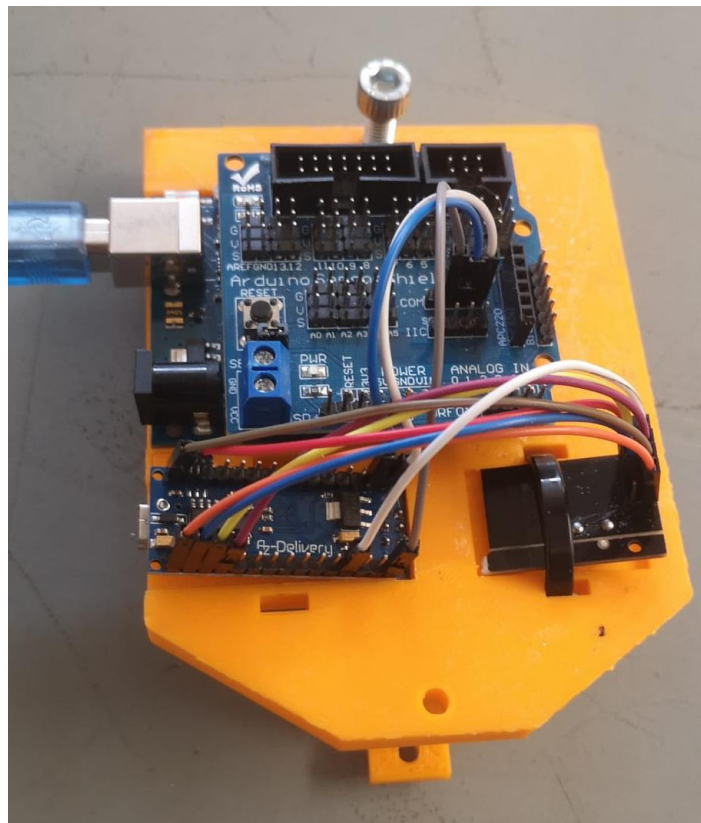


Figura 4. 75. Ensamblaje soporte superior

2. **Ensamblaje del soporte inferior** [Figura 4.76 y 4.77]. Los elementos que ha acoplar en el serán:
- Pantalla LCD.
 - Arduino uno para pantalla y sensor de distancia.
 - Bomba y válvula para las ventosas.
 - Soporte para sistema de alimentación.

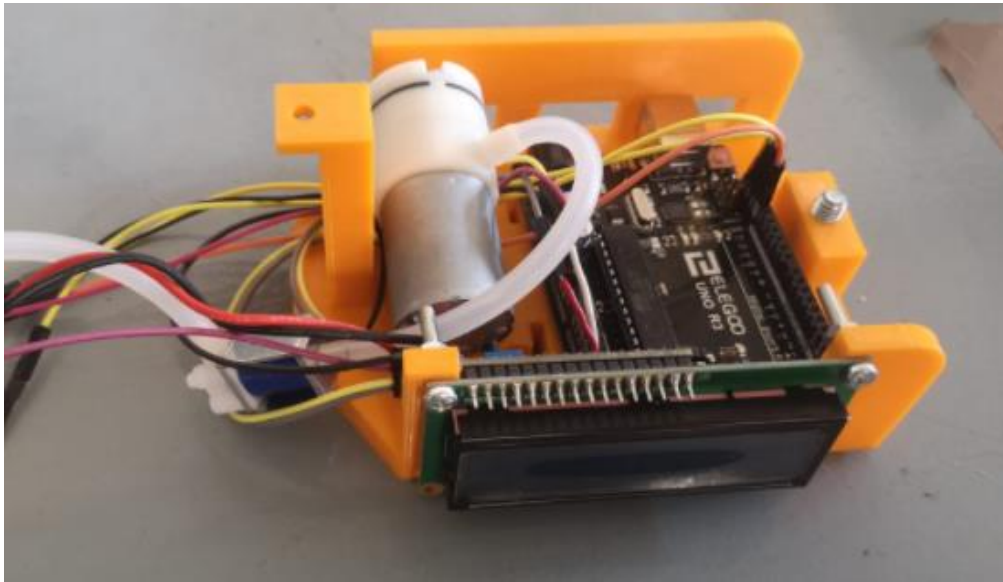


Figura 4. 76. Ensamblaje soporte inferior

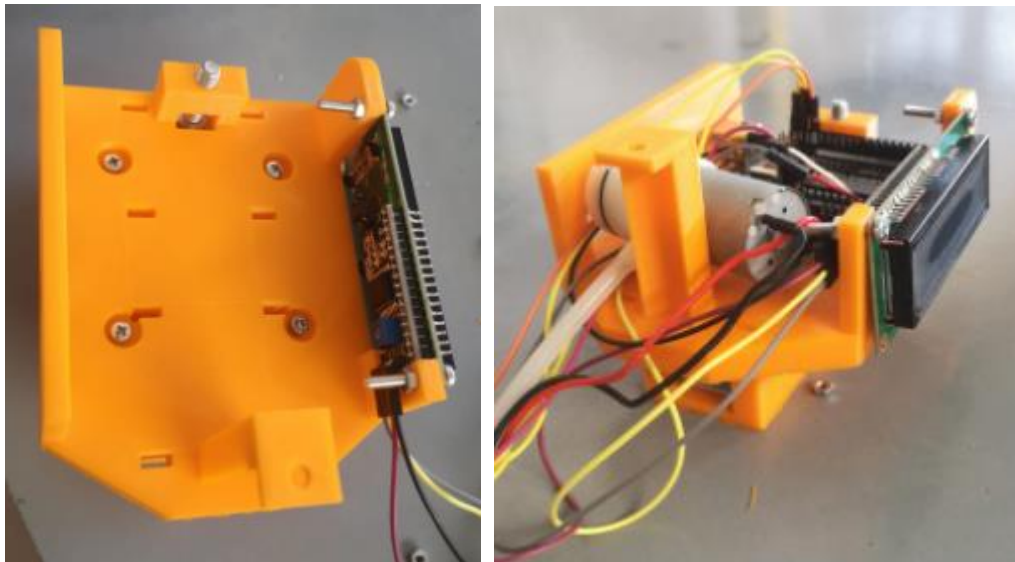


Figura 4. 77. Detalle ensamblaje soporte inferior más soporte para alimentación

3. **Base completa (superior e inferior)** [Figuras 4.78 y 4.79]. Los elementos que ha acoplar en ella serán:
- Servomotor
 - Mecanismo 4 barras (pinza)
 - Sistema de ventosas

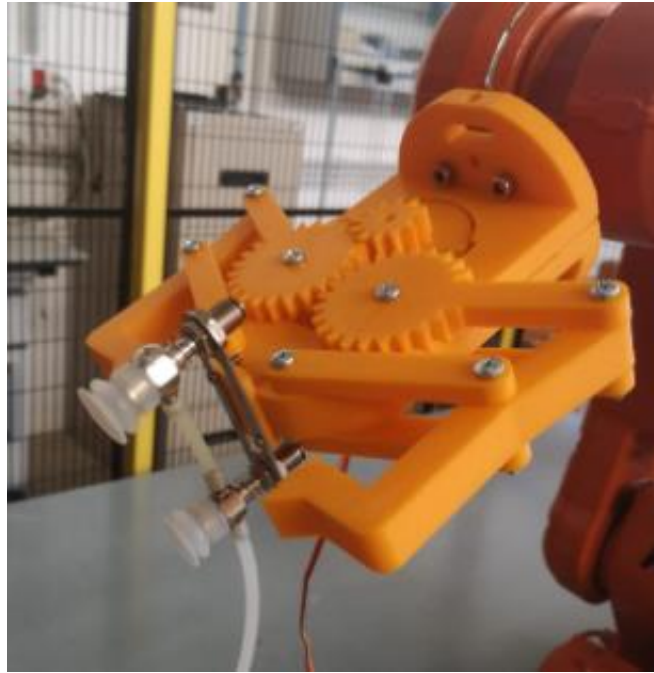


Figura 4. 78. Ensamblaje bases (Vista superior)

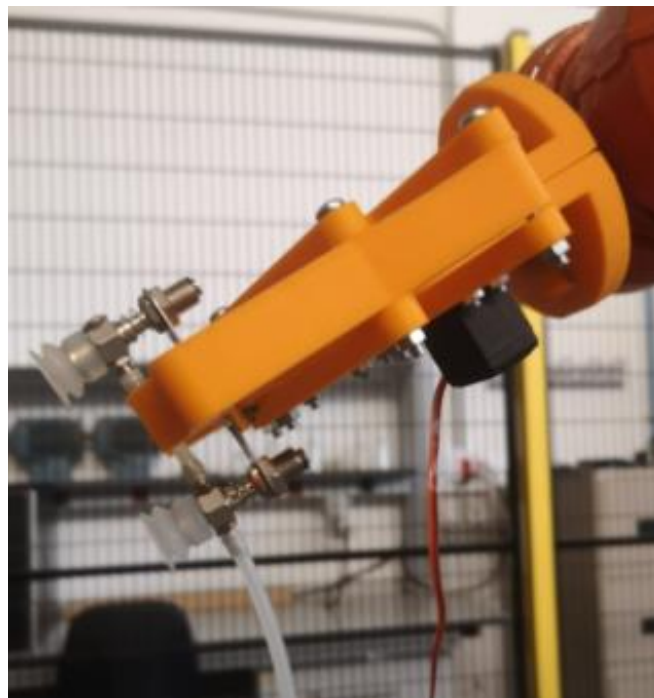


Figura 4. 79. Ensamblaje bases (vista inferior)

4. **Acoplamiento base y soporte inferior** [Figura 4.80]. Incluye todos los elementos que cada uno de los ensamblajes mostrados.

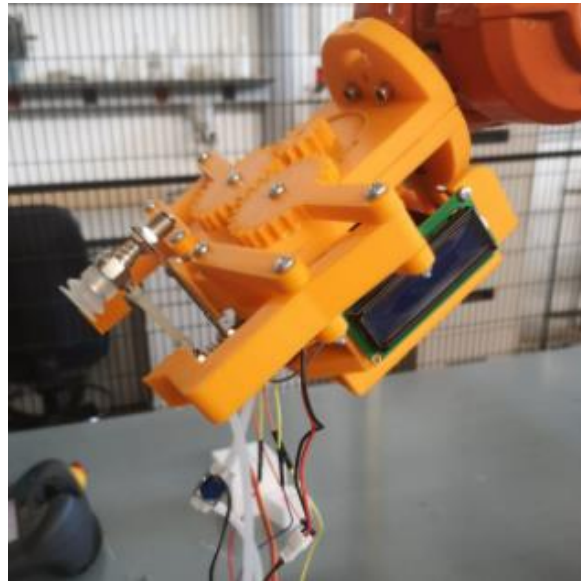


Figura 4. 80

5. **Herramienta pinza-ventosa completa** [Figura 4.81 y 4.82]. Al ensamblaje anterior se le incluye también la base superior y el acoplamiento con el sensor de distancia.



Figura 4. 81. Ensamblaje herramienta completa (Vista izquierda)

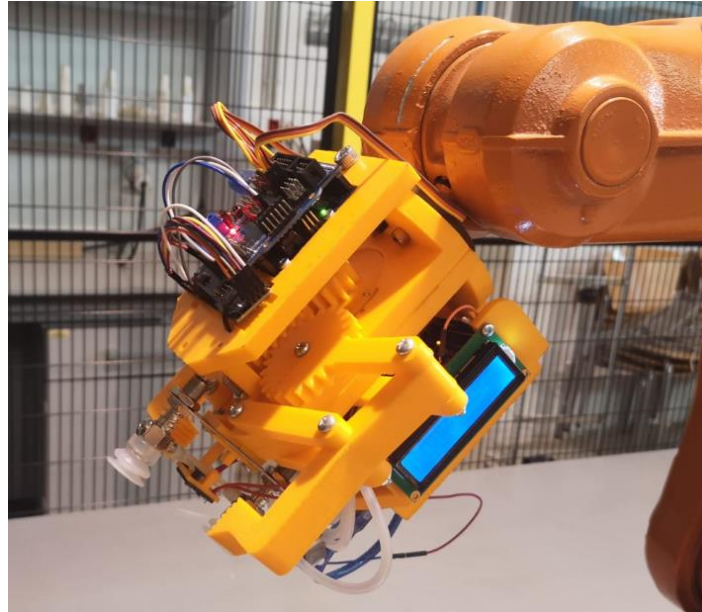


Figura 4. 82. Ensamblaje herramienta completa (Vista derecha)

Dentro de este montaje debemos destacar el acoplamiento realizado al sensor de distancia, ya que debido a la necesidad de que este elemento no sobresaliese por delante de las ventosas, se ha decidido soldar el cableado directamente a los pines necesarios y su soporte se ha diseñado con una apertura de manera que se puedan sacar por la parte posterior los cables necesarios [Figura 4.83]:

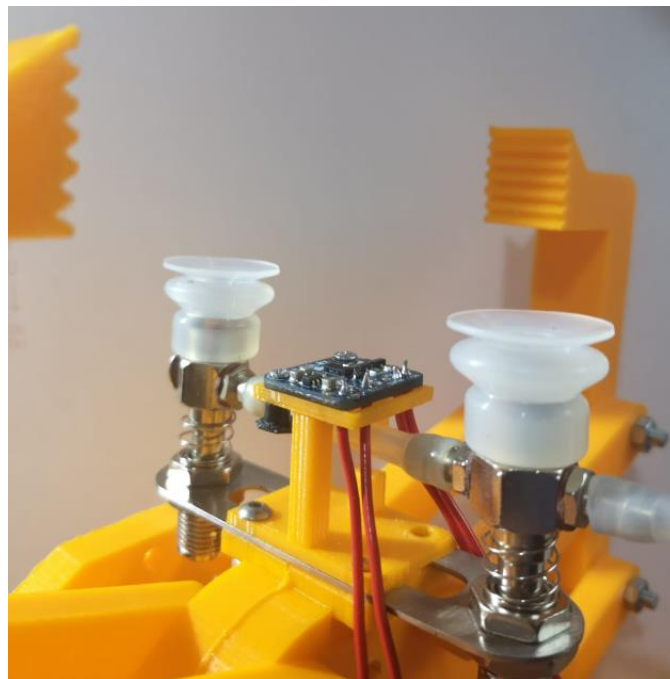


Figura 4. 83. Detalle conexión sensor de distancia

6. Sistema completo Robot-Herramienta. [Figura 4.84]

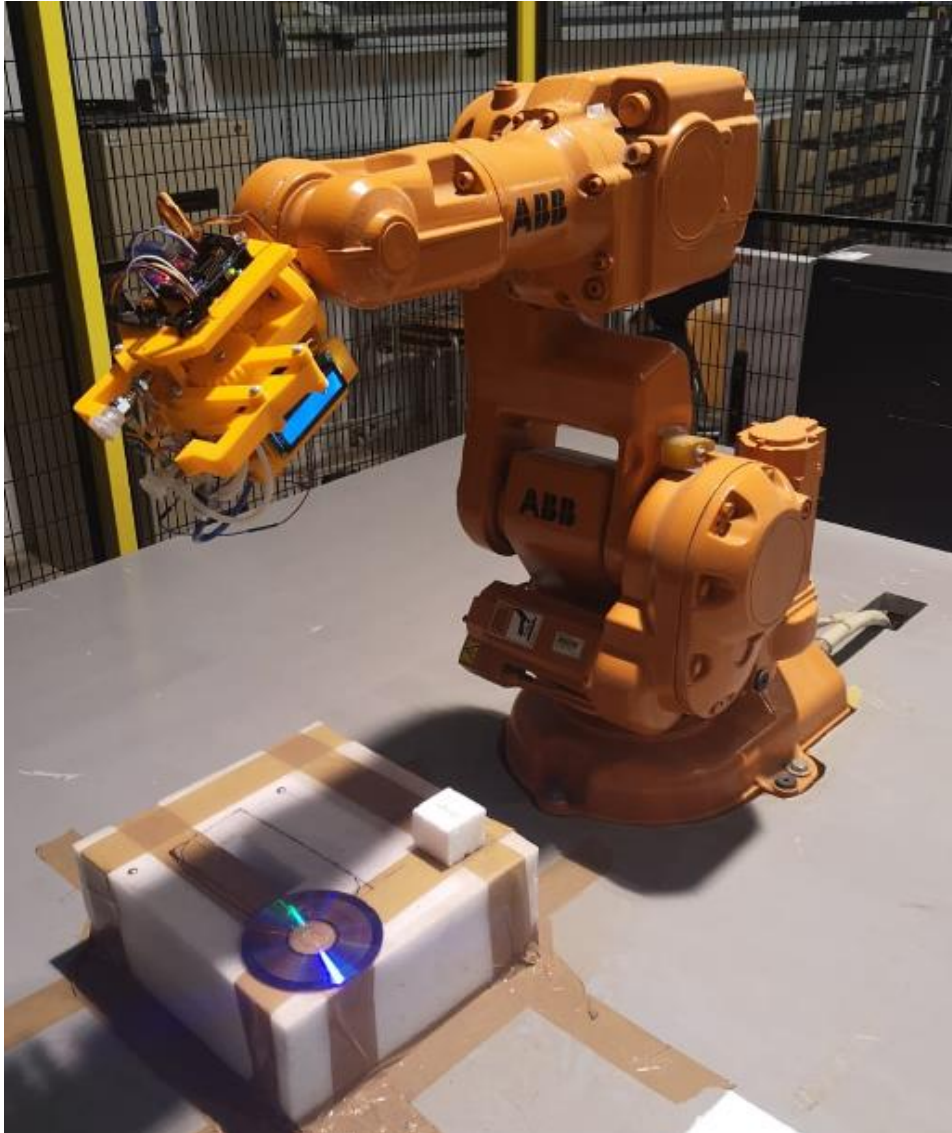


Figura 4. 84. Ensamblaje Sistema Robot-Herramienta completo.

Tras el montaje completo de la herramienta, podemos afirmar que el resultado es satisfactorio puesto que no se producen interferencias entre los distintos elementos y se adecua al modelo diseñado mediante métodos virtuales.

Otro punto a comprobar para asegurar la validez del diseño es la capacidad del Robot ABB IRB 140 de trabajar con ella. Para ello la herramienta ha de pesar menos de 6kg. Tras el proceso de pesaje de nuestra herramienta [Figura 4.85] hemos obtenido un peso de 927g, por lo que se podrá afirmar que el resultado es satisfactorio y nuestro robot no tendrá ningún problema para trabajar con la herramienta.

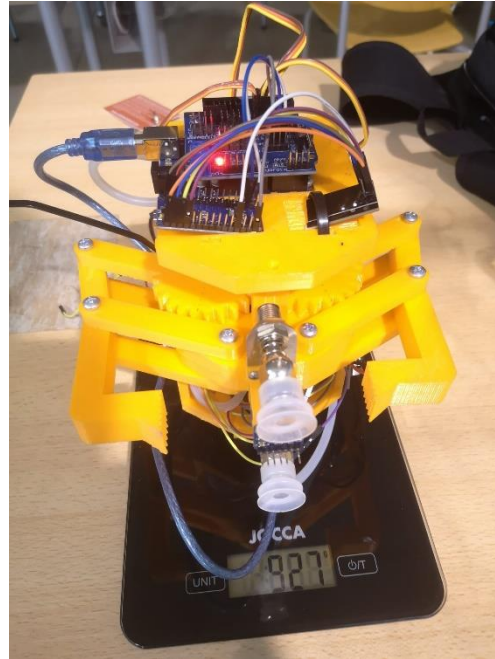


Figura 4. 85. Peso herramienta completa

4.5. PROGRAMACIÓN DE LAS FUNCIONALIDADES DE LA HERRMIENTA

Una vez se han conseguido físicamente cada una de las piezas que forma la herramienta, se procede a la implementación del sistema electrónico, encargado del movimiento y el control del conjunto herramienta-robot.

Las funciones que se deben conseguir a través del sistema electrónico incluido en este proyecto son:

- La configuración de la herramienta debe permitir manipular piezas mediante dos sistemas distintos: sistema de pinzas y sistema de agarre por ventosas.
- Se diseñará la herramienta para que a través del controlador IRC5C del propio robot y un conjunto de placas Arduino sea capaz de ejecutar las instrucciones de manipulación de elementos. Se proporcionará así al sistema la capacidad de operar de forma autónoma o mediante un sistema de control programable por un operario.
- Las herramientas utilizadas para ello y sus modelos son las expuestas en el punto anterior.

En este apartado se mostrará el esquema general de funcionamiento del conjunto, así como la funcionalidad y programación de cada uno de los componentes de la parte electrónica.

Se mostrará también el modo de comunicación entre los componentes para alcanzar el cumplimiento de todos los requisitos de diseño.

4.5.1. ELEMENTOS NECESARIOS

Los elementos incluidos en el funcionamiento del sistema electrónico de control ya se han mencionado anteriormente. Sin embargo, para una mayor claridad de su diseño, en este punto se podrán encontrar tabulados todos los elementos necesarios para la realización del sistema de manipulación y control de nuestro proyecto.[Tabla 22]

COMPONENTE	UNIDADES
Controlador IRC5C del robot	1
Arduino Uno	2
Arduino Nano	2
Sensor Shield v5.0	1
Módulo de comunicación NRF24L01	2
Servomotor. [<i>Modelo Power HD - Servomotor Standard - HD-3001HB</i>]	1
Sistema de ventosas con bomba de succión. [Kit para 25T Servos MG996 MG995 DS3218]	1
Pantalla Lcd y módulo de comunicación I2C	1
Sensor de distancia VL53L0X	1
ProtoBoard	1
Interruptores	6
Led	6
Alimentación	1
Cableado de conexión	

Tabla 22. Elementos necesarios para la programación de funcionalidades

4.5.2. ESQUEMA DE FUNCIONAMIENTO CONJUNTO. DIAGRAMA DE CONEXIONES

A través de programación RAPID mediante RobotStudio, se dotará al controlador IRC5C del programa necesario para la manipulación del robot hasta los distintos puntos de trabajo. Este programa también será el encargado de proporcionar las salidas digitales que activarán el funcionamiento de la herramienta pinza-ventosa acoplada en la muñeca del robot.

Las señales enviadas por el controlador del robot serán recibidas por el Arduino Nano (1) que se encuentra cableado al módulo de salidas propio del robot. Este Arduino, programado mediante su entorno propio Arduino IDE, será el encargado de detectar la salida procedente del controlador del robot y, mediante el módulo de comunicación NRF24L01, las transmitirá al Arduino nano (2) que se encuentra acoplado a la herramienta pinza-ventosa.

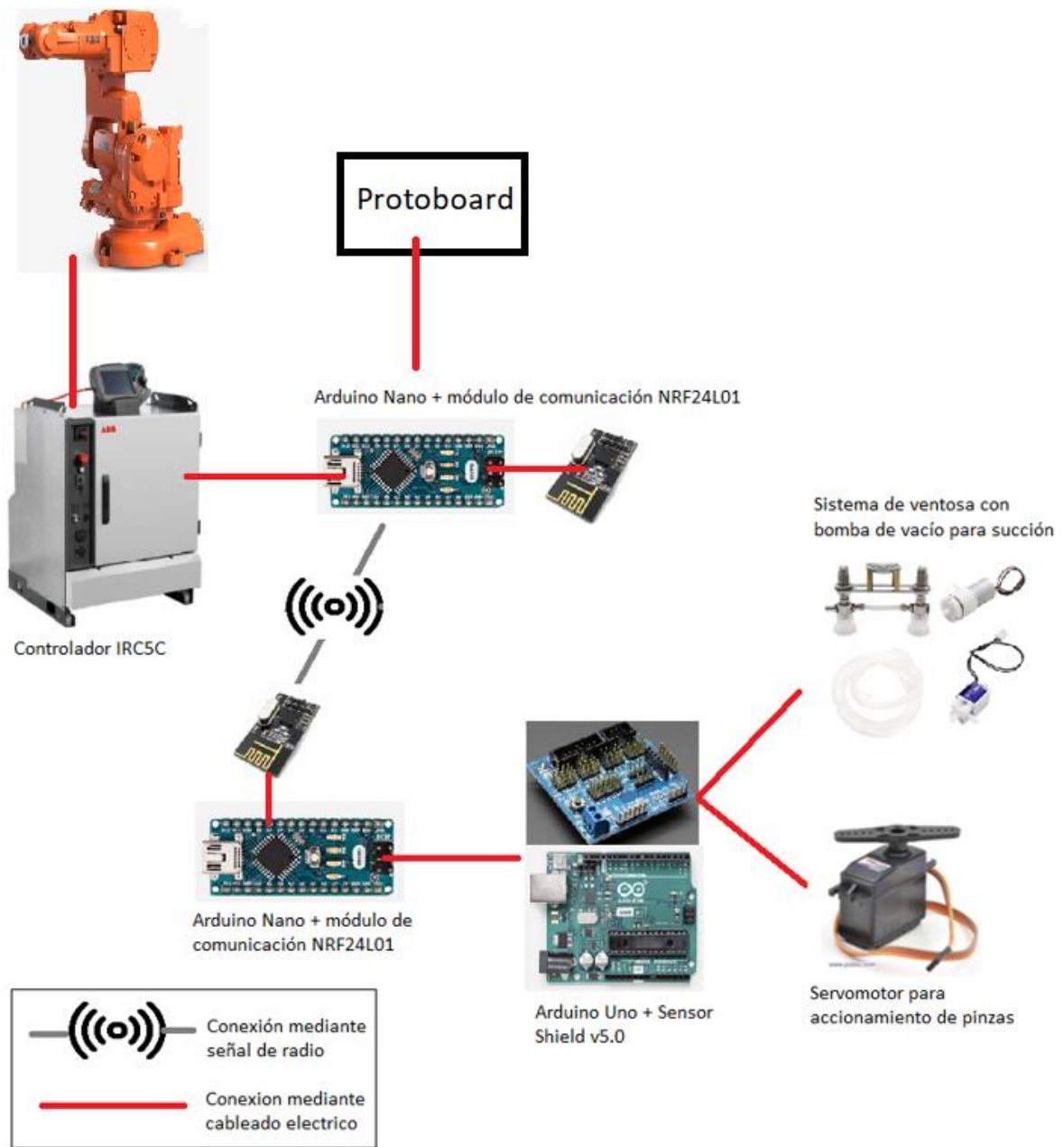
Una vez el Arduino Nano (2) ha recibido las señales procedentes del Arduino Nano (1) acoplado al controlador del robot, transmitirá estas instrucciones a través del puerto serie al Arduino Uno junto al que está cableado.

El Arduino Uno incluido dentro de la herramienta tendrá la tarea de recibir y leer las señales a través de su puerto serie y en función de ellas dar las instrucciones correspondientes para activar el sistema de pinzas o las ventosas. Este Arduino estará equipado con el sensor Shield v5.0 para facilitar esta tarea de control de los actuadores.

De esta manera el conjunto del sistema será capaz de, a través de las instrucciones enviadas inicialmente por el controlador del robot, operar todo el conjunto de la herramienta y hacerlo trabajar para el cumplimiento de las especificaciones de este proyecto.

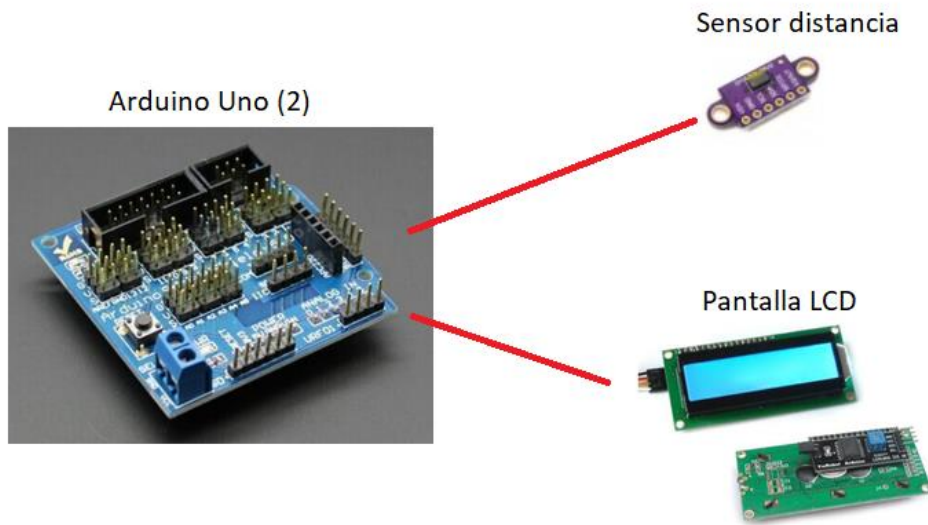
Para una mayor flexibilidad de nuestro sistema, se ha diseñado adicionalmente una placa que podrá simular el funcionamiento del controlador del robot. De esta manera, se podrá operar la herramienta de forma manual a través de una serie de interruptores que equivaldrán a las salidas del controlador, es decir, que un operario externo será capaz de controlar el sistema pinza-ventosa de manera manual e independiente.

Se muestra a continuación un esquema explicativo del proceso descrito anteriormente [Esquema 2]:



Esquema 2. Diagrama de conexiones del sistema de control

Adicionalmente, se ha incluido otro subsistema electrónico encargado de operar el dispositivo de medición de distancia y la pantalla LCD que se encuentran dentro del ensamblaje de la herramienta [Esquema 3]. El funcionamiento de este subsistema es sencillo, se ha programado el Arduino Uno (2) para que lea la distancia proporcionada por el sensor de distancia (distancia desde el frontal de la herramienta hasta el elemento más próximo) y lo muestre a través de la pantalla.



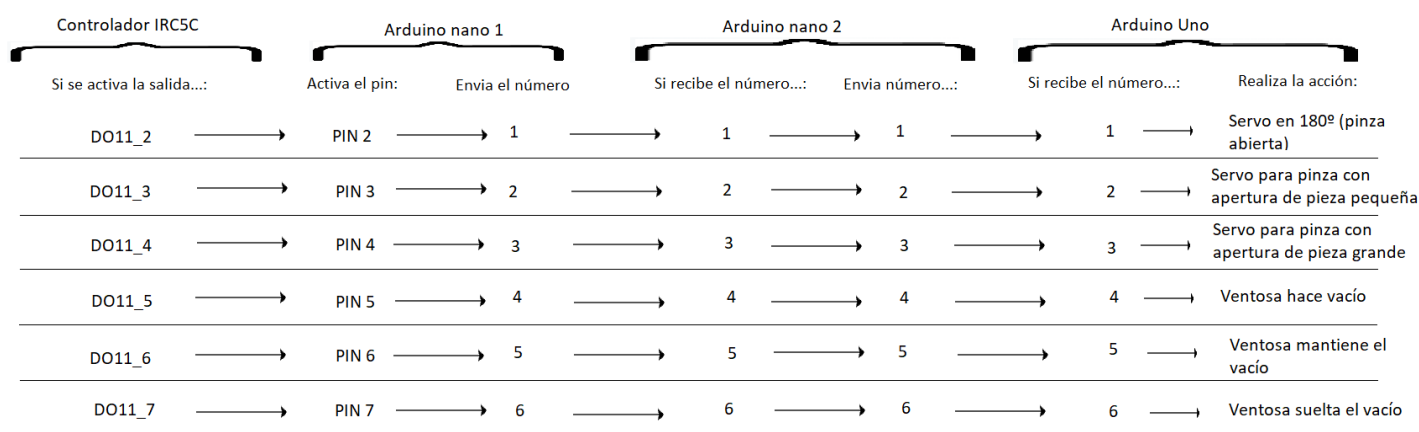
Esquema 3. Diagrama de conexiones del sistema sensor de distancia-pantalla

Todo este sistema estará alimentado a través de una batería portátil de 5V que irá acoplada a la herramienta.

De manera muy esquemática, se podría explicar el funcionamiento del sistema de control de la pinza y la ventosa en los siguientes pasos:

1. En primer lugar, el controlador del robot activará una de sus salidas (DO11_2, DO11_3...) dependiendo del código implementado en él.
2. Esta salida estará conectada a uno de los pines del Arduino nano 1, el cual se activará. Al activar este pin, mediante la programación de este Arduino, se enviará un número que dependerá del pin activado, por ejemplo, el pin 2 hará que se envíe el número 1, el pin 3 hará que se envíe el número 2 y así sucesivamente. Estos números serán enviados y recibidos mediante los módulos de radio.
3. El número correspondiente llegará al Arduino nano 2, el cual lo leerá y lo transmitirá mediante el puerto serie al Arduino Uno con el que está cableado.
4. Una vez el número llegue al Arduino Uno, el cual está conectado con los actuadores de la pinza y la ventosa, este realizará una u otra acción dependiendo del número que le llegue.

El esquema general de este proceso es el que puede encontrarse de manera resumido en la siguiente imagen [Esquema 4]:



Esquema 4. Esquema general de salidas/Entradas para el funcionamiento del ciclo de control

En el siguiente apartado se podrá entrar en mayor detalle en la programación de cada una de las partes involucradas en el proceso.

4.5.3. PROGRAMACIÓN DE LOS DISTINTOS COMPONENTES

Dentro de este apartado se mostrará la programación correspondiente a cada uno de los elementos que forman parte del control electrónico de nuestro sistema. Se detallarán también las funciones que realiza cada uno de ellos dentro del conjunto.

Como ya se ha explicado anteriormente, la programación de los sistemas se realizará a través del software propio de Arduino llamado Arduino IDE, a excepción de la programación del controlador del robot que se ha realizado en lenguaje RAPID a través del Software de ABB RobotStudio.

4.5.3.1. PROGRAMACIÓN MEDIANTE ROBOT STUDIO

A través del software Robot Studio se programará en lenguaje RAPID el controlador IRC5C de ABB que será el encargado de controlar el movimiento del brazo robótico y enviar las señales para el funcionamiento de los distintos actuadores incorporados en la herramienta pinza-ventosa.

El proceso correspondiente a la simulación y obtención de la programación en RobotStudio se ha englobado dentro del punto 4.6. SIMULACION DEL SISTEMA MEDIANTE ROBOTSTUDIO.

4.5.3.2. PROGRAMACIÓN MEDIANTE ARDUINO IDE

Dentro de nuestro proyecto tendremos 4 placas Arduino distintas, los cuales tendrán una programación distinta y estarán conectados con distintos componentes. Se programará el funcionamiento de cada uno de ellos a través del lenguaje propio de Arduino [6].

Para la explicación correspondiente a esta parte del funcionamiento del sistema, se partirá de las señales que envía el controlador IRC5C una vez programado y en funcionamiento. Estas señales activarán los distintos puertos módulo DSQC 653 de entradas y salidas incorporado en el control del robot, la cual estará cableado a mi sistema como se verá a continuación.

Se han distinguido 4 etapas correspondientes a los distintos controladores y programas que forman el sistema de control completo. Se explica a continuación la finalidad y el funcionamiento de cada una de ellas.

ETAPA 1: ARDUINO NANO 1 + MODULO COMUNICACIÓN (CONJUNTO EMISOR)

Una vez programado el controlador del robot como se explica en el punto 4.6, la siguiente etapa para conseguir el control de nuestro sistema es la programación del Arduino nano 1.

El controlador del robot activará las salidas dentro del módulo DSQC 653, estas salidas irán cableadas directamente a los distintos pines de mi Arduino nano 1 de la siguiente manera:

- Salida DO11_2 → pin 2
- Salida DO11_3 → pin 3
- Salida DO11_4 → pin 4
- Salida DO11_5 → pin 5
- Salida DO11_6 → pin 6
- Salida DO11_7 → pin 7

Estas conexiones harán que una vez se active la salida correspondiente, se active el pin correspondiente cableado a ella dentro la placa Arduino, pasando así a estado "HIGH"(1) [7].

A través de los distintos pines, se podrá desarrollar el programa correspondiente a esta fase de control.

Funcionamiento del programa para el Arduino nano 1

Dentro su entorno Arduino IDE, realizaremos el programa cuya función es la de detectar cuál de los pines está activo y en función de ello enviar un número distinto para cada caso al Arduino nano 2, que irá acoplado a la herramienta en la muñeca del robot.

Para conseguir esto, nuestro programa se ha estructurado de la siguiente manera:

1. Se incluyen todas las librerías necesarias:

```
#include<nRF24L01.h>
#include<RF24.h>
#include<RF24_config.h>
#include<SPI.h>
```

En este caso se han incluido las librerías correspondientes al módulo de radio nRF24L01 que será el responsable de transmitir el resultado a la siguiente fase.

2. Se inicia el módulo de radio

```
const int pinCE= 9;
const int pinCSN= 10;
RF24 radio(pinCE,pinCSN);
```

3. Se declaran una serie de variables de tipo entero con los pines de la placa Arduino. Estas variables reciben el nombre de la acción asociada en la que se traducirán en la etapa final de control de la herramienta.

```
const int inputPinza180 =2; //Pin que activa abrir pinza
const int inputPinza44 =3; //Pin que activa cerrar pinza a 44mm
const int inputPinza88 =4; //Pin que activa cerrar pinza a 88mm
const int inputVentosaSuc=5; //Pin que activa la succión de la
ventosa
const int inputVentosaAguanta=6; //Pin que activa que la ventosa
aguante la succion
const int inputVentosaSuelta=7; //Pin que activa soltar presión de la
ventosa
```

4. Se han declarado las variables tipo byte que posteriormente se enlazarán a estos pines. Estas variables solo recibirán 2 valores posibles, 0(Low) cuando el pin correspondiente este desactivado y 1(High) cuando se encuentren activos.

```
byte v2=0; //valor del pin 2 (correspondiente a abrir pinza)
byte v3=0; //valor del pin 3 (correspondiente a cerrar pinza 44mm)
byte v4=0; //valor del pin 4 (correspondiente a cerrar pinza 88mm)
byte v5=0; //valor del pin 5 (correspondiente a succion de la
ventosa)
byte v6=0; //valor del pin 6 (correspondiente a aguantar succion)
byte v7=0; //valor del pin 7 (correspondiente a fin succion)
```

5. Se termina de configurar nuestro puerto de radio y se crea la variable data en la cual se cargará el valor que se va a transmitir desde este Arduino hasta el que se encuentra en la herramienta:

```
// Single radio pipe addressforthe 2 nodestocommunicate.  
constuint64_t pipe = 0xE8E8F0F0E1LL;  
  
intdata[1];
```

6. Dentro del setup se configurarán nuestras variables como pines en modo entrada:

```
voidsetup(void)  
{  
  radio.begin();  
  radio.openWritingPipe(pipe);  
  pinMode(inputPinza180, INPUT);  
  pinMode(inputPinza44, INPUT);  
  pinMode(inputPinza88, INPUT);  
  pinMode(inputVentosaSuc, INPUT);  
  pinMode(inputVentosaAguanta, INPUT);  
  pinMode(inputVentosaSuelta, INPUT);  
}
```

7. Dentro del bucle de nuestro programa en primer lugar se le dará el valor 0 a la variable data, tras ello se asocian las variables v2 a v7 con sus pines correspondientes:

```
voidloop(void)  
{  
  data[0]= 0;  
  
  v2 =digitalRead(inputPinza180); //Lee la entrada del PIN 2 y lo  
  guarda en v2  
  v3 =digitalRead(inputPinza44); //Lee la entrada del PIN 3 y lo  
  guarda en v3  
  v4 =digitalRead(inputPinza88); //Lee la entrada del PIN 4 y lo  
  guarda en v4  
  v5 =digitalRead(inputVentosaSuc); //Lee la entrada del PIN 5 y lo  
  guarda en v5  
  v6 =digitalRead(inputVentosaAguanta); //Lee la entrada del PIN 6  
  y lo guarda en v6  
  v7 =digitalRead(inputVentosaSuelta); //Lee la entrada del PIN 7 y  
  lo guarda en v7
```

8. Tras ello, nos centramos en el grueso del programa. El programa, mediante un bucle if, comprobará los valores de los distintos pines y dependiendo de qué

condición se cumpla enviará un número que irá del 1 al 6. Si no se cumple ninguna de las condiciones el programa enviará un 0.

Dentro de la programación se han incluido los comentarios necesarios para comprender el código al completo.

Se enviará un valor cada 1 segundo (el tiempo se establece en milisegundos dentro de la programación).

```
if (v2 ==HIGH&& v3==LOW&& v4==LOW) { //Cuando la salida 2 esta
activa (y la 3 y 4 no) quiero abrir la pinza a 180° y envio un 1
  data[0]=1; //valor que se envia
  radio.write(data,sizeof data); //envio mediante radio
  delay(1000); //tiempo de espera para enviar un valor cada 1000
milisegundos
}

elseif(v3 ==HIGH&& v2==LOW&& v4==LOW){ //Cuando la salida 3 (y la
2 y 4 no) esta activa quiero cerrar la pinza a 44 y envio un 2
  data[0]=2;
  radio.write(data,sizeof data);
  delay(1000);
}

elseif(v4 ==HIGH&& v2==LOW&& v3==LOW){ //Cuando la salida 4 esta
activa (y la 3 y 2 no)quiero cerrar la pinza a 88 y envio un 3
  data[0]=3;
  radio.write(data,sizeof data);
  delay(1000);
}

elseif(v5 ==HIGH){ //Cuando la salida 5 esta activa Succion
con la ventosa, envio un 4
  data[0]=4;
  radio.write(data,sizeof data);
  delay(1000);
}

elseif(v6 ==HIGH){ // Cuando la salida 6 esta activa Aguanto
el vacio, envio un 5
  data[0]=5;
  radio.write(data,sizeof data);
  delay(1000);
}

elseif(v7 ==HIGH){ // Cuando la salida 7 esta activa Aguanto el
vacio, envio un 6
  data[0]=6;
  radio.write(data,sizeof data);
  delay(1000);
}
else{ //Si no, envio 0 que no repercuten en mi codigo
  data[0]=0;
  radio.write(data,sizeof data);
  delay(1000);
}
}
```

Para que esta fase de programación funcione, se ha cableado el Arduino nano de la manera que corresponde con el módulo de comunicación nRF24L01 [Figura 4.86]:

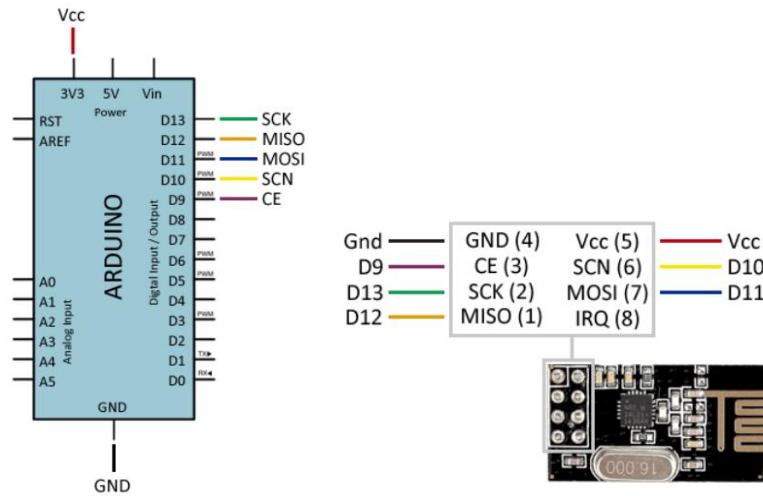


Figura 4. 86. Cableado Arduino Nano- nRF24L01

Para dotar a nuestro sistema de mayor flexibilidad se ha construido una placa utilizando la protoboard TW-E41-1060, mediante la cual se pueden activar los distintos pines manualmente mediante interruptores. Se ha incluido un led junto a cada uno de los interruptores para mayor visualización de las acciones realizadas. [Figura 4.87]

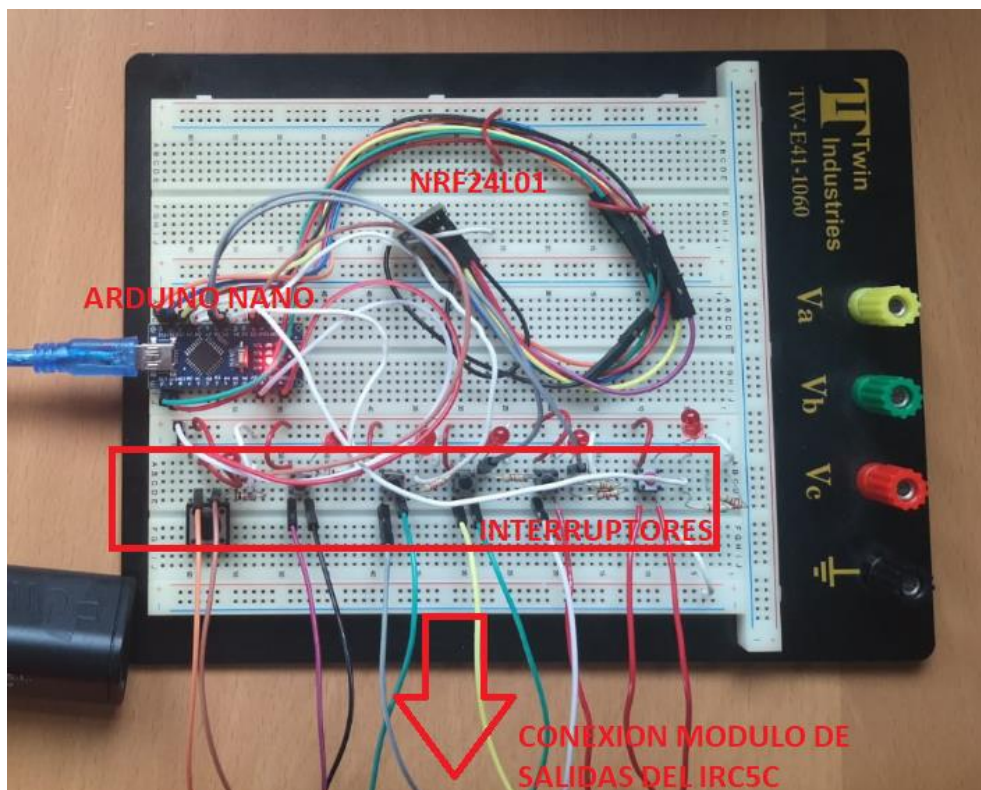


Figura 4. 87. Placa correspondiente a la primera fase de control mediante Arduino

De esta manera un operario externo podrá enviar órdenes a la herramienta de igual manera que lo hará el controlador del robot.

Esta placa se encontrará situada junto al controlador del robot ya que ha de ir cableada con el módulo de salidas de este.

ETAPA 2: ARDUINO NANO (2) + MODULO COMUNICACIÓN

El encargado de recibir las variables numéricas enviadas en el anterior paso será este Arduino nano (2), que se encuentra situado en la herramienta colocada en la muñeca del robot, sobre el soporte superior [Figura 4.88]:



Figura 4. 88. Colocación del Arduino Nano dentro del ensamblaje

La misión de este Arduino dentro del control del sistema es la de recibir las variables enviadas por el Arduino nano (1) que se encuentra junto con el controlador y transmitírselas al Arduino Uno que controla los actuadores de las pinzas y las ventosas.

Para llevar a cabo esta tarea se ha realizado el siguiente programa:

1. En primer lugar, incluimos las bibliotecas necesarias para trabajar con el módulo de comunicación de radio nRF24L01 y el puerto serie. Se configuran también los pines del puerto de radio.

```
#include<SoftwareSerial.h>
#include<nRF24L01.h>
#include<RF24.h>
#include<RF24_config.h>
#include<SPI.h>

const int pinCE= 9;
const int pinCSN= 10;
RF24 radio(pinCE,pinCSN);

// Single radio pipe address for the 2 nodes to communicate.
const uint64_t pipe = 0xE8E8F0F0E1LL;
```

2. Se crea la variable data donde se almacenarán los datos recibidos mediante la comunicación de radio del nRF24 procedentes del Arduino nano 1. Se inician también ambos puertos de comunicación:

```
int data[1];

void setup() {
  Serial.begin(9600);
  radio.begin();
  radio.openReadingPipe(1, pipe);
  radio.startListening();
}
```

3. Dentro del bucle del programa, si hay un valor disponible del puerto de radio se lee y se guarda en la variable data, posteriormente se envía esta variable data mediante el puerto serie:

```
void loop()
{
  if (radio.available())
  {
    radio.read(data, sizeof data);
    Serial.print(data[0]);
  }
  delay(1000);
}
```

Se enviará un dato cada 1 segundo.

Se podrá realizar una comprobación de que los valores enviados son los esperados mediante la visualización del puerto serie conectando el Arduino nano al ordenador y activando los pines manualmente mediante la ProtoBoard como se explicó anteriormente [Figura 4.89 y 4.90]:



Figura 4. 89. Visualización del puerto serie



Figura 4. 90. Resultados obtenidos de la visualización del puerto Serie

El receptor de esta variable “data” será el Arduino uno que se explicará a continuación, el cual este cableado junto a este Arduino mediante puerto serie.

Para el cableado mediante puerto serie se ha seguido la siguiente configuración [Figura 4.91]:

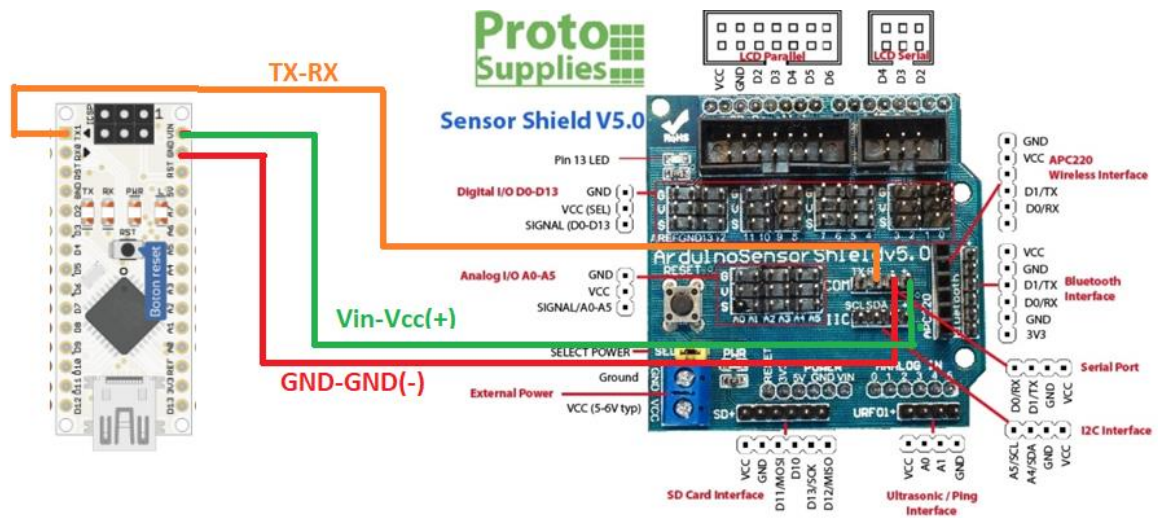


Figura 4. 91. Cableado puerto serie Arduino Nano- Arduino Uno

FASE 3: ARDUINO UNO (1) + SENSOR SHIELD

La variable “data” transmitida por el anterior Arduino nano es leída ahora por el Arduino uno cableado a él mediante su puerto serie.

Este Arduino uno se encuentra sobre la herramienta y será el encargado de controlar los actuadores del sistema pinza y de las ventosas. [Figura 4.92]



Figura 4. 92. Situación Arduino Uno sobre la herramienta

El programa encargado del control de este Arduino constará de las siguientes fases:

1. En primer lugar, se incluye la librería necesaria para el control de los servomotores y se declaran las variables correspondientes a cada uno de los servos presentes:
 - Servo1 → corresponde al de movimiento del mecanismo de pinza
 - Servo2 → corresponde a la bomba que realiza la succión en las ventosas
 - Servo3 → corresponde a la válvula para mantener y soltar el vacío de las ventosas

```
#include<Servo.h>

// Declaramos la variable para controlar el servo
Servo servo1;
Servo servo2;
Servo servo3;
```


2. Posteriormente, se crea la variable “dato” donde se guardará el valor leído por el puerto serie proveniente del anterior Arduino nano (variable “data” de la etapa anterior).

```
int dato;
```

3. Dentro del setup de nuestro programa se inician los pines donde irán configurados cada uno de los servomotores, se inicia la comunicación con el puerto serie y se establece el valor inicial 0 para cada uno de los servomotores.

```
void setup() {  
  
  //Inicio los pines donde van los distintos servos  
  servo1.attach(6); //-> Servo pinza  
  servo2.attach(5); // -> Bomba  
  servo3.attach(4); // ->Valvula  
  
  //Inicio la comunicacion con el puerto serie  
  Serial.begin(9600);  
  
  //Pongo todo en su posicion 0  
  servo1.write(180); //Pinza abierta  
  servo2.write(0); //Ventosa sin presion  
  servo3.write(0); //Ventosa sin presion  
}
```

4. Dentro del bucle principal de nuestro programa, en primer lugar, se lee la variable que nos llega desde el puerto serie (“data”) y se carga dentro de “dato”, se comprueba la variable dato y mediante una estructura tipo case, el Arduino realizará unas acciones u otras en función del valor de la variable dato:

- Lectura y comprobación de la variable dato.

```
void loop() {  
  dato=Serial.read();  
  switch(dato){ //Comprobamos el dato  
}
```

- Si el valor de dato es igual a 1, el Arduino abrirá la pinza hasta su máxima apertura. Para ello se coloca el servo 1 a 180°.

```
  case '1': //Si recibimos un '1' abro total la pinza...  
    servo1.write(180);  
    break;
```

- El caso en el que dato = 2: Se cerrará la pinza para para coger la pieza pequeña. Contando que esta pieza tendrá unas dimensiones de 44 mm de lado, se ha calculado que la apertura necesaria del servomotor correspondiente es de 42, 1º. Estos cálculos podrán encontrarse dentro del anexo de cálculos correspondiente.

```
case'2': //si recibimos una '2' cierro la pinza a 44
mm para coger pieza pequeña
servo1.write(42,1);
break;
```

- El caso en el que dato = 3: Se cerrará la pinza para para coger la pieza grande. Esta pieza tendrá unas dimensiones de 88 mm, se ha calculado que la apertura necesaria del servomotor correspondiente es de 99, 58º (cálculos dentro del anexo de cálculos correspondiente).

```
case'3': //si recibimos una '3' cierro la pinza a
44mm para coger pieza pequeña
servo1.write(99,58);
break;
```

- El caso en el que dato = 4: Significará que se desea coger una pieza haciendo uso de las ventosas, para ello se realiza succión mediante la bomba del sistema de ventosas (servo2.write(180)). La válvula que mantiene la presión permanecerá cerrada (servo3.write(0)) y la pinza en el caso de que se quiera usar la ventosa deberá encontrarse totalmente abierta para no interferir con ella (servo1.write(180)).

```
case'4': //si recibimos una '4' Ventosa hace vacio
servo2.write(180);
servo3.write(0);
//EN TODAS ESTAS LA PINZA DEBE ESTAR ABIERTA DEL
TODO, no molestar a la ventosa
servo1.write(180);
break;
```

- El caso en el que dato = 5: Significará que se desea manipular la pieza que se ha cogido haciendo uso de las ventosas, para ello se mantendrá la succión mediante la válvula del sistema de ventosas. La válvula permanecerá cerrada (servo3.write(0)) y ya no se hará succión mediante la bomba (servo2.write(0)) la pinza deberá encontrarse totalmente abierta para no interferir con el sistema de ventosas (servo1.write(180)).

```
        case '5': //si recibimos una '5' la ventosa Aguantar
el vacío
            servo2.write(0);
            servo3.write(0);
            servo1.write(180);
            break;
```

- Por último, si dato=6 la ventosa soltará el vacío. Para ello el servo de la válvula se abrirá (servo3.write(180)). Los servos de la bomba y la pinza se encontrarán en el mismo estado que el caso anterior.

```
        case '6': //si recibimos una '6' la ventosa suelta el
vacío
            servo2.write(0);
            servo3.write(180);
            servo1.write(180);
            break;
    }
}
```

Mediante esta programación, nuestro sistema será capaz de controlar las distintas acciones de la herramienta pinza-ventosa.

FASE 4: ARDUINO UNO (2)

Por último, se ha incluido esta fase de control, la cual no estará interconectada con la explicada anteriormente.

En ella, a través de un Arduino uno, podremos conocer la distancia que hay desde el frontal de la herramienta hasta el elemento más cercano y mostrar esta distancia por una pantalla LCD acoplada a la base de la herramienta.

Para este proceso ha sido necesario un nuevo Arduino Uno ya que la comunicación con el sensor se realiza mediante puerto serie y las dos placas Arduino ya introducidas en la herramienta tienen este puerto ocupado. El puerto i2c tampoco estaría disponible para realizar la comunicación mediante él ya que será necesario para conectar la pantalla LCD. La conexión del Arduino y ambos sistemas queda de la siguiente manera [Figura 4.93]:

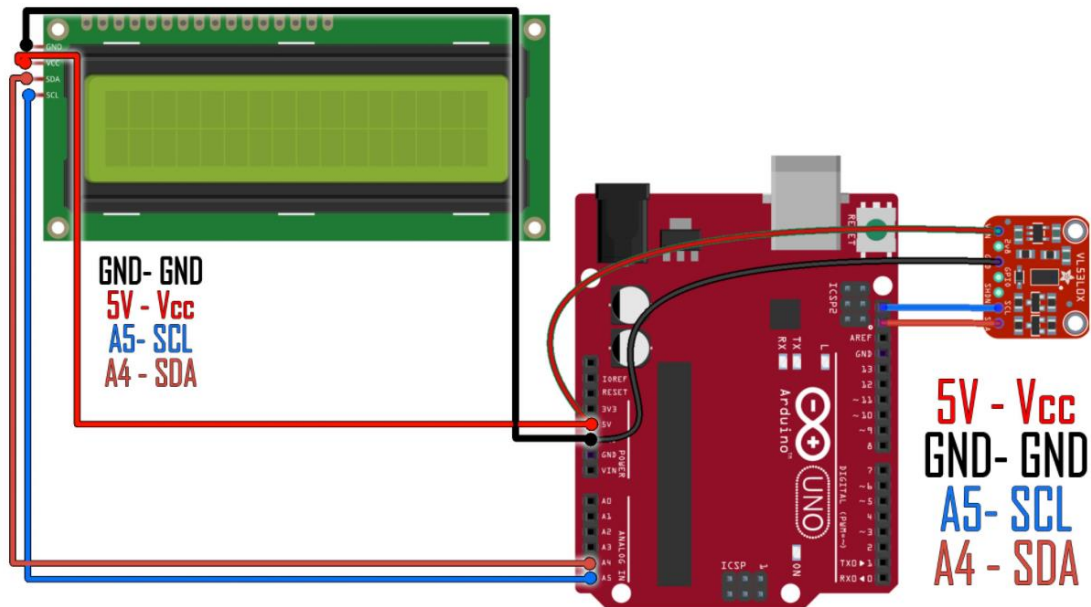


Figura 4. 93. Conexión sistema pantalla y sensor de distancia

Para llevar a cabo la tarea de leer y mostrar los datos de distancia de este conjunto se ha realizado el siguiente código:

1. Incluimos en primer lugar las librerías *Adafruit_VL53L0X.h* (necesaria para el sensor de distancia), *Wire.h* y *LiquidCrystal_I2C.h* (pantalla LCD)

```
#include"Adafruit_VL53L0X.h"
#include<Wire.h>
#include<LiquidCrystal_I2C.h>
```

2. Iniciamos el dispositivo LCD con sus parámetros correspondientes y se realiza una primera inicialización de la pantalla y la lectura del sensor de distancia.

```
// Configuración de los caracteres de pantalla
LiquidCrystal_I2C lcd(0x3F, 16, 2);

Adafruit_VL53L0X lox = Adafruit_VL53L0X();

void setup() { //inicio puertos de la pantalla y puerto serie
  lcd.init();
  lcd.backlight();
  Serial.begin(9600);

  // Esperar puerto serie
  while (!Serial) {
    delay(1);
  }
}
```

```
Serial.println("Test Adafruit VL53L0X");
if (!lox.begin()) {
  Serial.println(F("Fallo boot VL53L0X"));
  while(1);
}
// power
Serial.println(F("VL53L0X API Simple Ranging example\n\n"));
}
```

3. Dentro del bucle del programa, se leerá la medida del sensor de distancia y dependiendo del rango en el que se encuentre, se mostrarán por pantalla los distintos mensajes correspondientes. La medida de distancia se actualizará cada 100 milisegundos:

```
void loop() {
  VL53L0X_RangingMeasurementData_t measure;

  Serial.print("Leyendo medidas... ");
  lox.rangingTest(&measure, false); // 'verdadero' para obtener la
  impresión de los datos de depuración

  if (measure.RangeStatus != 4) {
    Serial.print("Distancia (mm): ");
    Serial.println(measure.RangeMilliMeter);
    lcd.clear(); //
    lcd.setCursor(0,0);
    lcd.print(" DISTANCIA en mm ");
    lcd.setCursor(0,1);
    lcd.print(measure.RangeMilliMeter);
  } else {
    Serial.println(" Fuera rango ");
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" DISTANCIA en mm ");
    lcd.setCursor(0,1);
    lcd.print(" Fuera rango ");
  }

  delay(100);
}
```

4.6. SIMULACION DEL SISTEMA MEDIANTE ROBOTSTUDIO

Dentro de este punto se mostrará el proceso completo de simulación del sistema mediante RobotStudio [8]. Esto incluirá todo el proceso desde la creación de la estación virtual hasta la programación y comprobación de las funcionalidades del sistema.

4.6.1. CREACIÓN DE LA ESTACIÓN Y EL ENTORNO DE TRABAJO

El primer paso será la creación de una nueva estación vacía dentro de RobotStudio, en la cual se insertará nuestro modelo de robot IRB 140 y su controlador IRC5C. A este controlador se le añadirá la tarjeta de entradas y salidas DSQC 653 ya que es la que se encuentra dentro de la instalación de nuestro robot en la Universidad de Almería.

Tras ello se procederá al estudio del programa ejemplo que se va a llevar a cabo para simular el funcionamiento de nuestro sistema robot junto con el uso de la herramienta diseñada.

El programa que se va a desarrollar deberá ser capaz de realizar los siguientes movimientos:

1. Coger una pieza cuadrada de 88 mm de lado y 80 mm de altura (pieza grande) desde un punto de la mesa del robot y trasladarla hacia otro punto.
2. Coger un disco desde un punto de la mesa del robot y trasladarla hacia otro punto.
3. Coger una pieza cuadrada de 44 mm de lado y 40 mm de altura (pieza pequeña) desde un punto de la mesa del robot y trasladarla hacia otro punto.

Para el manejo de las distintas piezas se ha de tener en cuenta la plataforma elevada que se encuentra frente al robot dentro de las instalaciones de la Universidad de Almería. Esta plataforma tendrá unas dimensiones de 330x300x125 mm como se muestra en la siguiente imagen [Figura 4.94]:

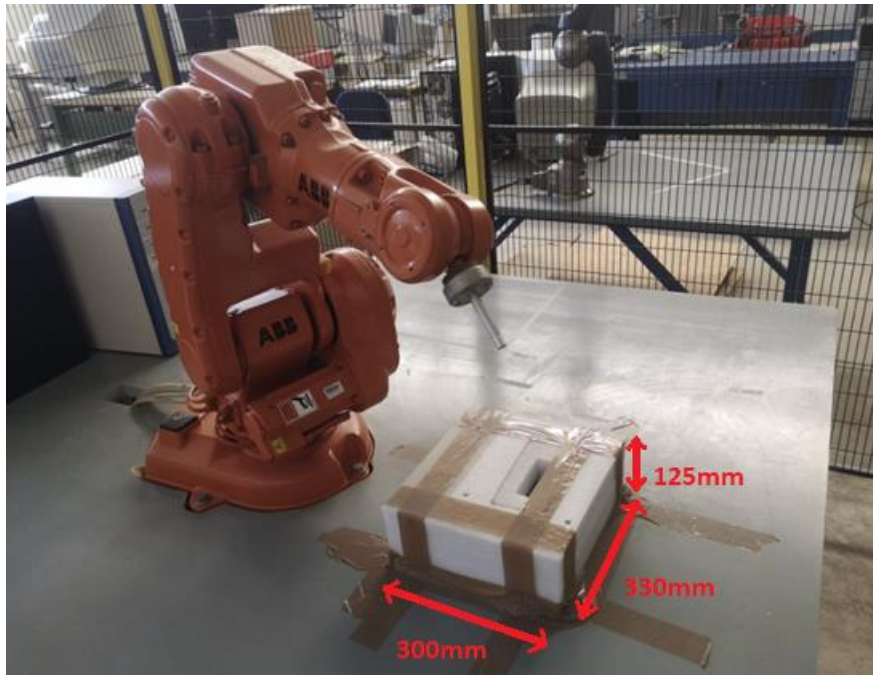


Figura 4. 94. Entorno del Robot en el taller de la UAL

Se ha decidido que la colocación de las piezas dentro de la mesa de trabajo del robot sea de la siguiente manera [Figura 4.95]:

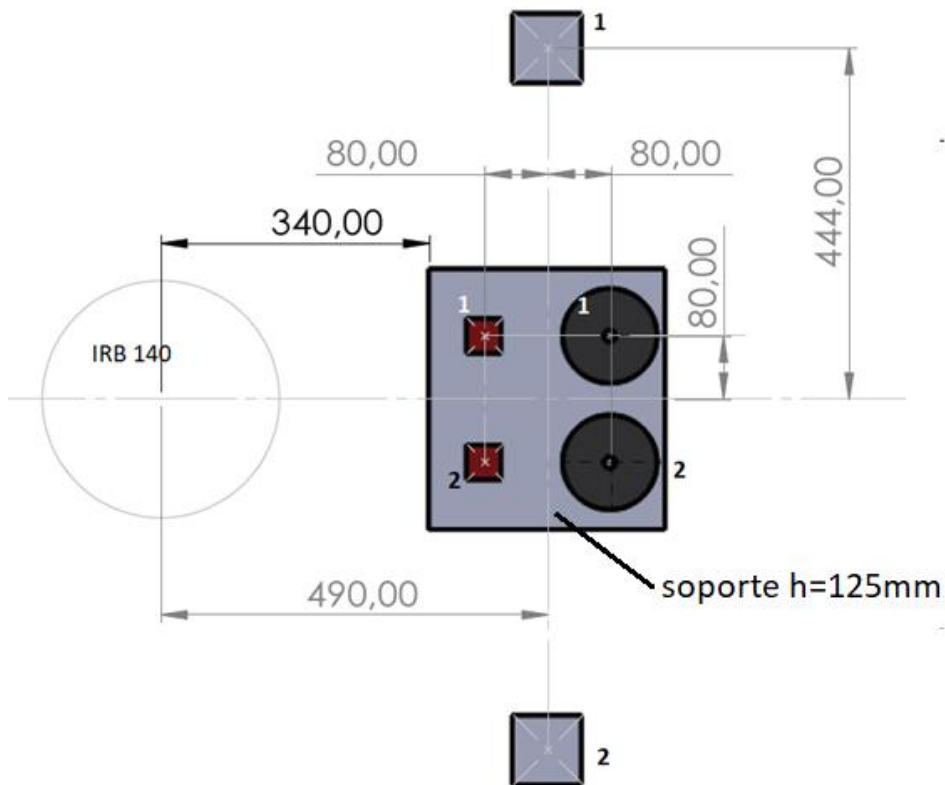


Figura 4. 95. Localización de los elementos en el entorno de trabajo del robot

Dentro de la imagen anterior se pueden observar las piezas marcadas con un 1 que indicarán la posición de estas al inicio del programa y las marcadas con un 2 que muestran la posición final donde se depositarán mediante la ejecución del programa.

Estas posiciones, referenciadas respecto al punto base del robot, tendrán las siguientes coordenadas [Figura 4.96]:

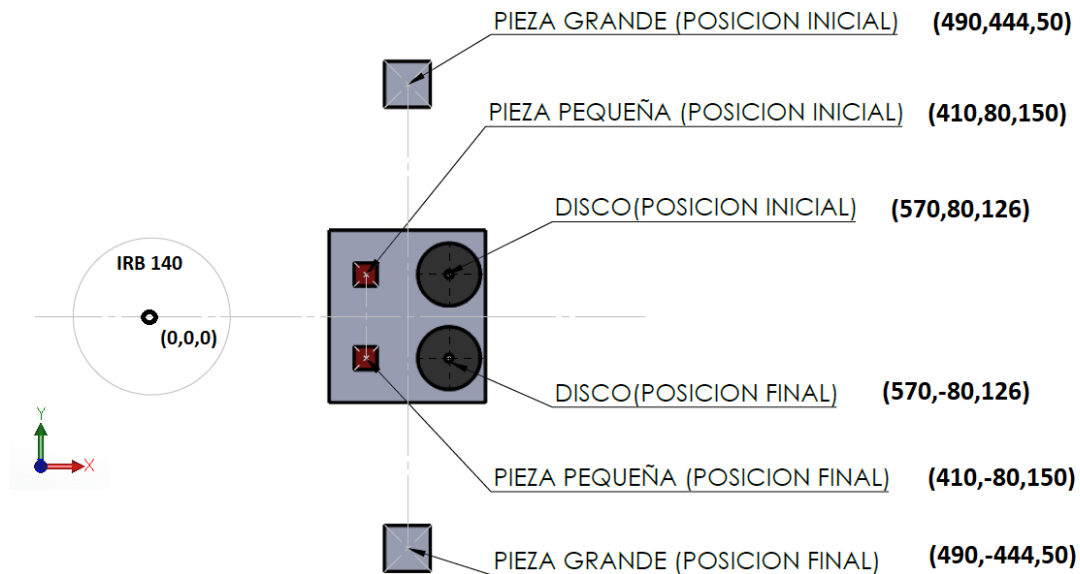


Figura 4. 96. Coordenadas de los distintos elementos del entorno de trabajo del robot

Una vez está claramente definido nuestro programa de ejemplo a solucionar a través de programación mediante RobotStudio, procederemos a configurar la estación. Para ello se siguen los pasos que se verán a continuación.

4.6.2. CONFIGURACIÓN DE LA HERRAMIENTA

El primer paso será el de incluir la herramienta dentro de su posición. Para ello se exportará el modelo 3D de la herramienta creado en SolidWorks a RobotStudio. Se ha necesitado transferir el archivo al formato STL a fin de una correcta lectura por parte de RobotStudio.

Una vez introducida la geometría, se la definirá como herramienta. Para ello se ha introducido su peso aproximado de 1kg y se ha incluido su TCP, que será el punto dentro de ella que se tomará como referencia para la realización de todas las acciones por parte del conjunto robot-herramienta. Es decir, este punto será como la punta de la herramienta.

Para nuestro diseño, el TCP de la herramienta se ha colocado en el punto medio entre las 2 ventosas [Figura 4.97]. De esta manera, se tomará como referencia el plano en el que se encuentra el efector final de las ventosas a la hora de programar la cercanía a un objeto.

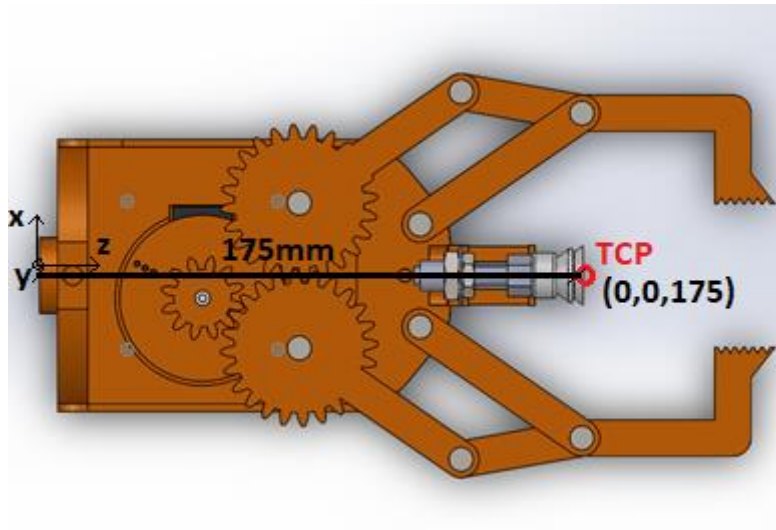


Figura 4.97. TCP herramienta

Una vez configurada la herramienta se acoplará al robot y se colocará en su correcta orientación. [Figura 4.98]

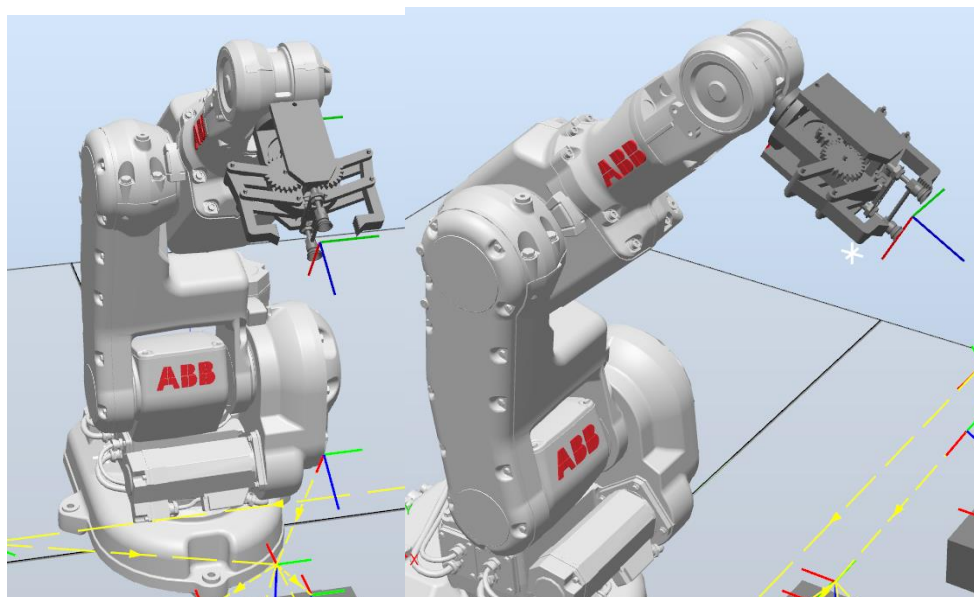


Figura 4.98. Acople herramienta-robot

En la siguiente imagen se puede ver el TCP de la herramienta respecto del cual actuará el sistema robot-herramienta en su posicionamiento frente a los puntos que se programen. [Figura 4.99]

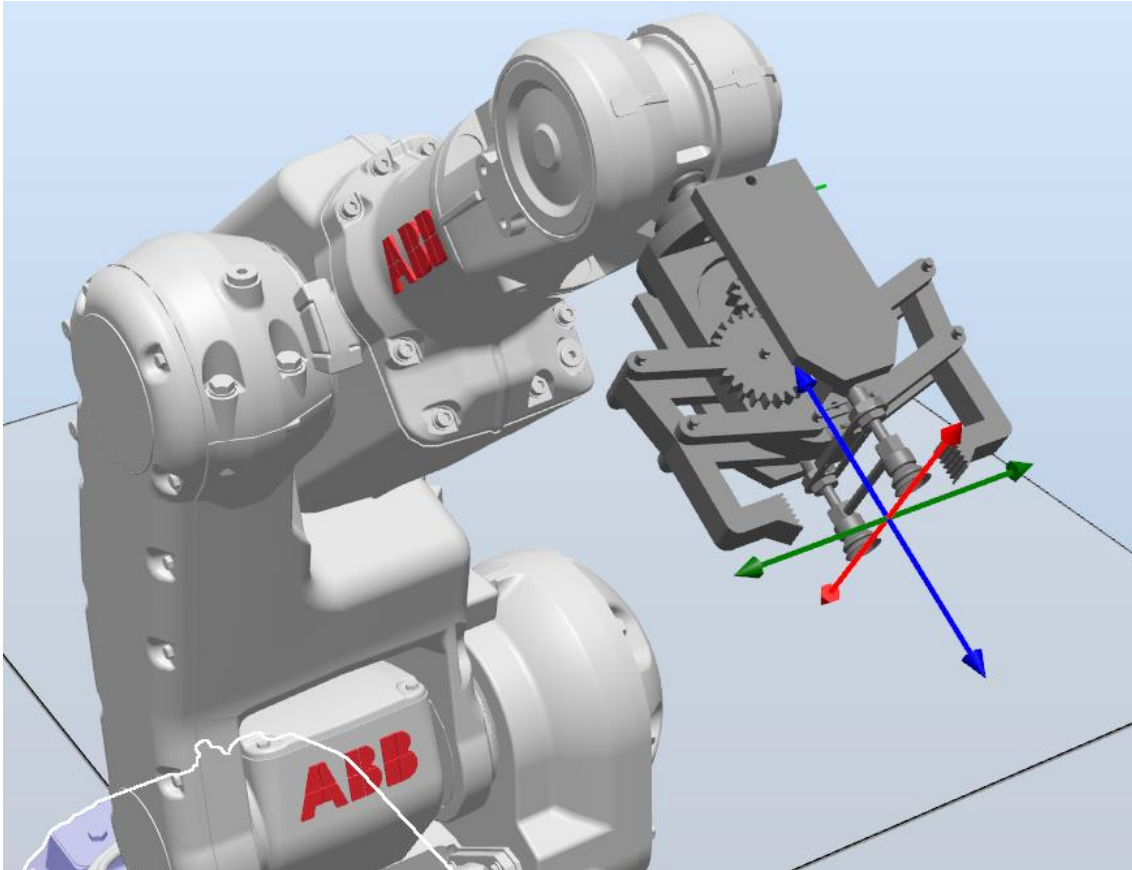


Figura 4. 99. TCP herramienta-robot

Tras ello se deberán crear las trayectorias a ejecutar por el robot dentro de nuestro programa.

4.6.3. CREACION DE LAS TRAYECTORIAS

El primer paso será crear un sistema de coordenadas de referencia respecto al cual estarán asociados los distintos puntos. A este sistema de referencia se le denomina WorkObject (wobj) dentro del lenguaje de RobotStudio.

Se creará el wobj respecto al punto central de la plataforma que se encuentra frente al robot. Con ello se conseguirá una mayor flexibilidad en la programación. Esto es debido a que, si una vez definidos los punto respecto esta base, se decidiera moverla, los puntos también se desplazarían con respecto a la nueva posición de la base.

Denominaremos a este WorkObject como wobj_base y se definirán todos los puntos de nuestro programa respecto a él, a excepción del punto de inicio correspondiente a la posición inicial de la herramienta. [Figura 4.100]

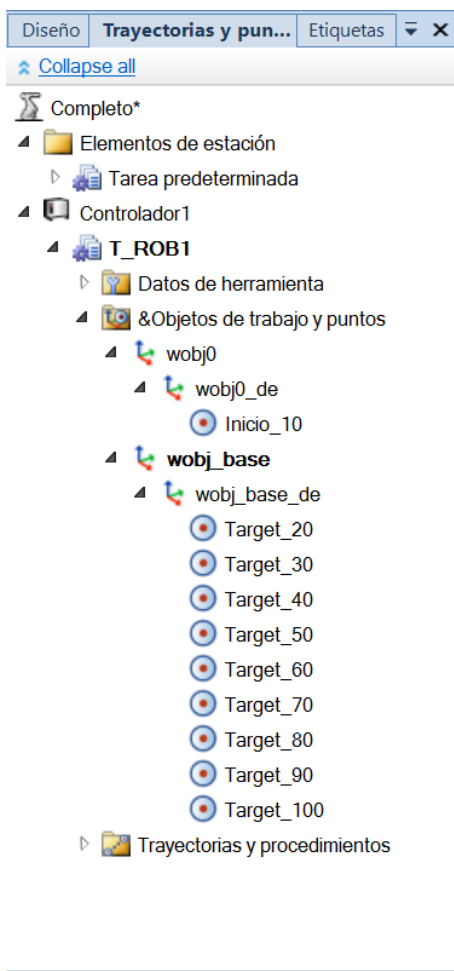


Figura 4. 100. Puntos creados dentro del entorno RobotStudio

Para definir los distintos puntos a los que ha de desplazarse el nuestro sistema se tendrán en cuenta las posiciones especificadas en la tabla 23. Se añadirán adicionalmente posiciones intermedias para que en los desplazamientos del robot no exista la posibilidad de colisión con otros objetos.

Target_20	posición dejar disco
Target_30	posición de desplazamiento Intermedia
Target_40	posición de coger disco
Target_50	posición de coger pieza grande
Target_60	posición de dejar pieza grande
Target_70	posición de desplazamiento Intermedia
Target_80	posición de desplazamiento Intermedia
Target_90	posición de coger pieza pequeña
Target_100	posición de dejar pieza pequeña

Tabla 23. Correlación puntos creados con la posición correspondiente

Para las posiciones que involucran la actuación de la pinza (coger y dejar piezas pequeña y grande). Se ha tenido en cuenta el offset adicional que se le ha de dar a la distancia desde el TCP de la herramienta hasta el punto donde se hará contacto con la pieza.

Es por ello por lo que en la coordenada z de los puntos relacionados con la manipulación de la pieza grande se han añadido 45 mm adicionales y en la de manipulación de la pieza pequeña se han añadido 25 mm. En las figuras 4.101 y 4.102 puede observarse el desarrollo de los milímetros que se han añadido a cada una de las posiciones.

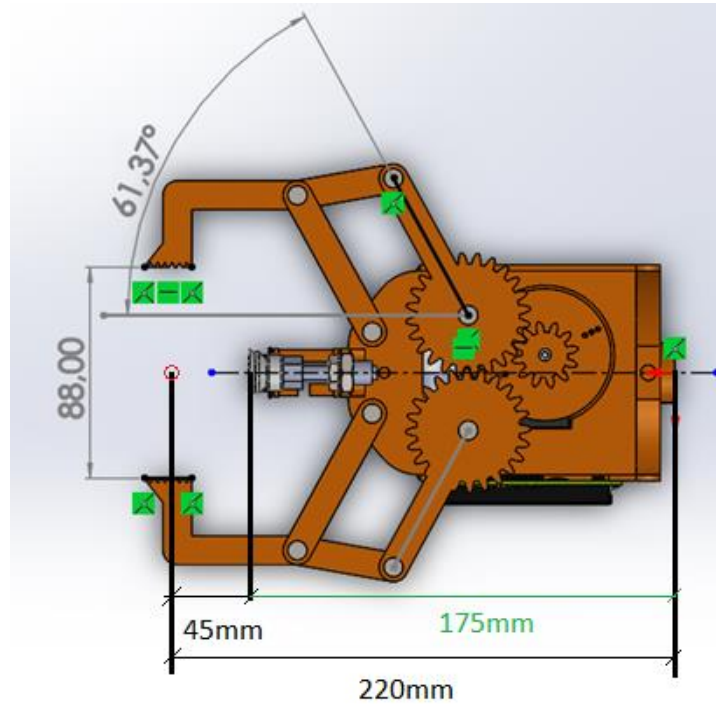


Figura 4. 101. Punto referido a la manipulación de piezas tipo grande

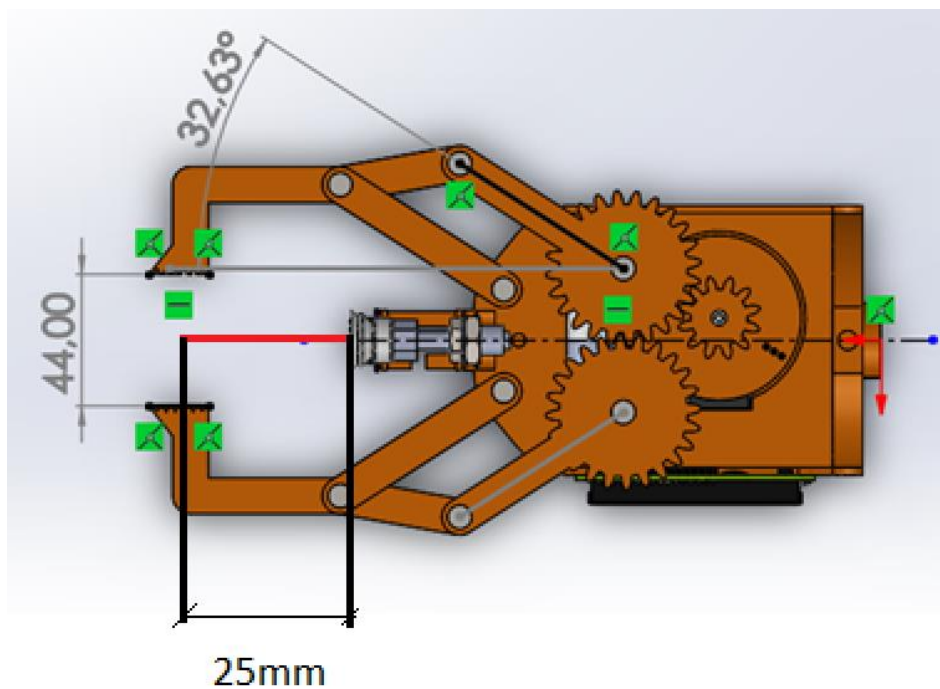


Figura 4. 102. Punto referido a la manipulación de piezas tipo pequeña

Se introducirá también la orientación correcta de la herramienta en cada uno de los puntos y se comprobará su correcto alcance para llegar a ellos. [Figura 4.103]

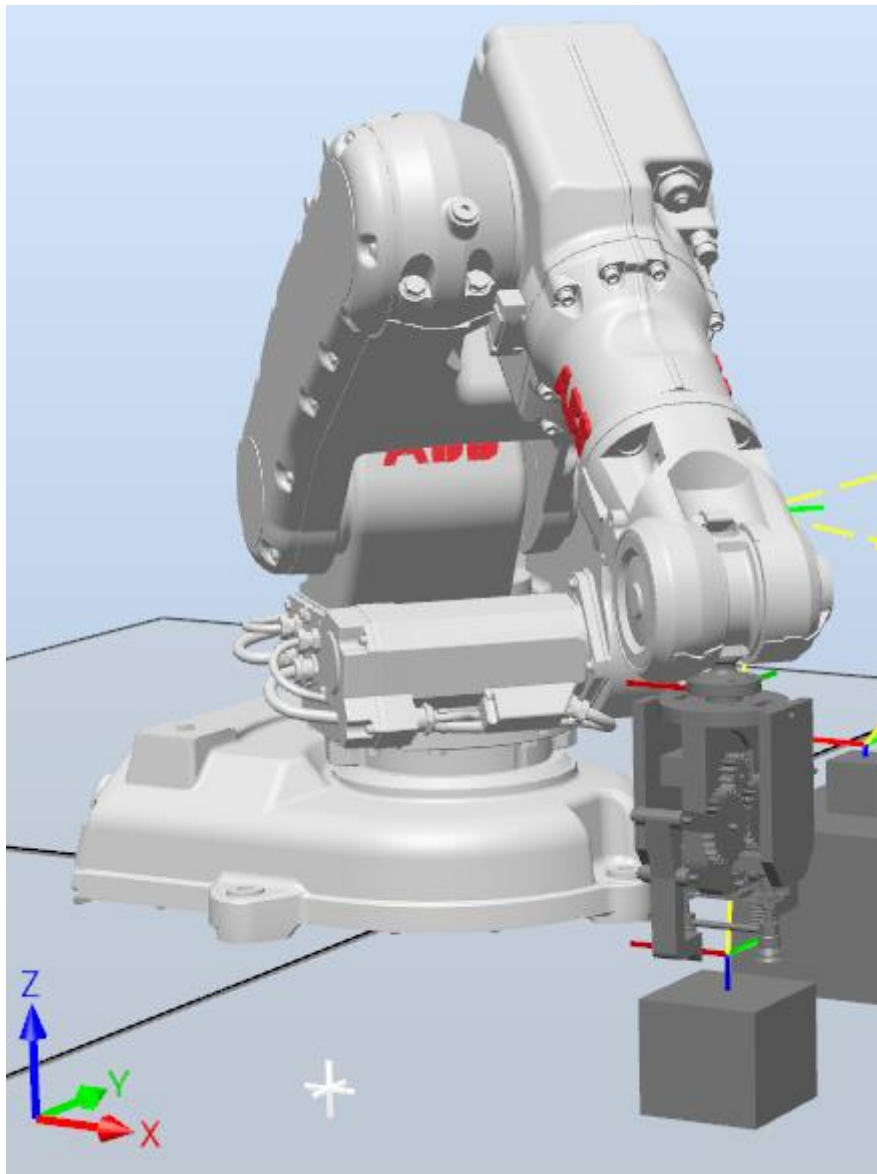


Figura 4. 103. Orientación de la herramienta en el punto de trabajo

Una vez se han creado todas las posiciones, estas se combinan para crear las distintas trayectorias que ha de recorrer el robot [Figura 4.104]:

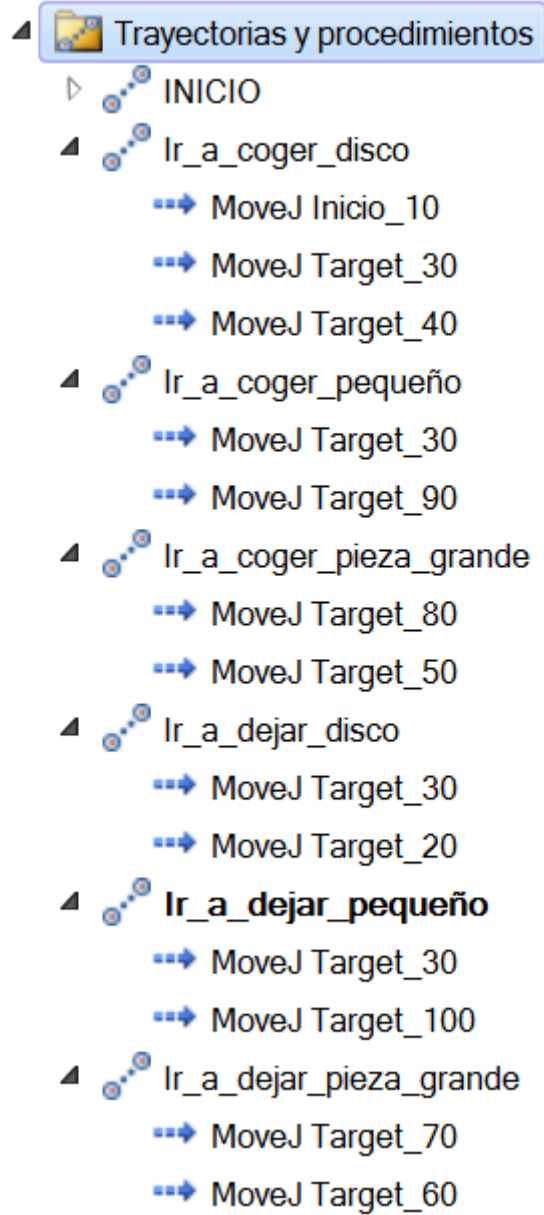


Figura 4. 104. Trayectorias

El movimiento seleccionado para los desplazamientos del robot es de tipo MoveJ, ya que en ellos será el controlador del robot el que seleccione el método óptimo para aproximarse a estos puntos.

4.6.4. CONFIGURACIÓN DEL CONTROLADOR

Posteriormente se configurarán las correspondientes señales de salida dentro de la tarjeta DSQC 653 incluida en el cuadro de control del robot [Figura 4.105].

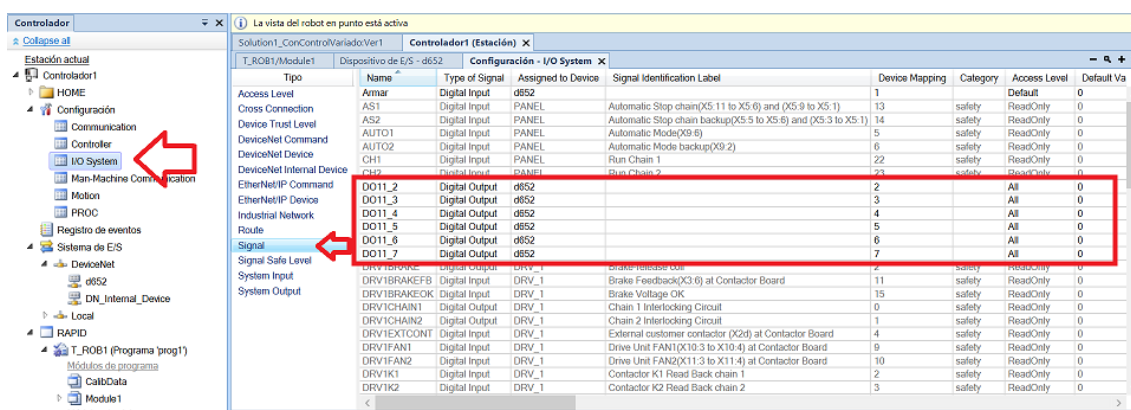
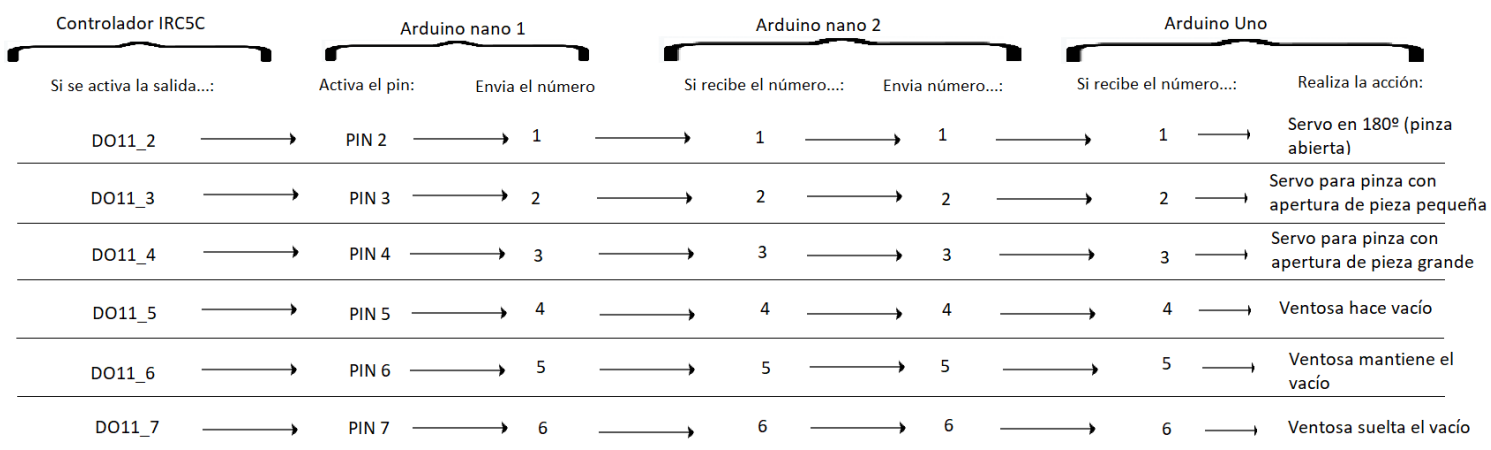


Figura 4. 105. Tarjeta DSQC 653 de salidas

Es a través de estas salidas (que se podrán activar o desactivar mediante programación) como se transmitirán las instrucciones a la siguiente etapa de control. Este proceso ya se explicó mostró de forma resumida en el apartado 4.5.2. Se puede apreciar la acción que desencadenará la activación de cada una de estas señales en el siguiente esquema 5:



Esquema 5. Ciclo de acción según la salida del controlador activa

Adicionalmente, dentro del controlador se ha incluido la lógica de la estación para obtener el movimiento de la pinza junto con las piezas a mover. Para ello, dentro de la lógica de la estación se ha asociado cada una de las salidas correspondientes a los movimientos de apertura y cierre de la pinza con su acción correspondiente. [Figura 4.106]

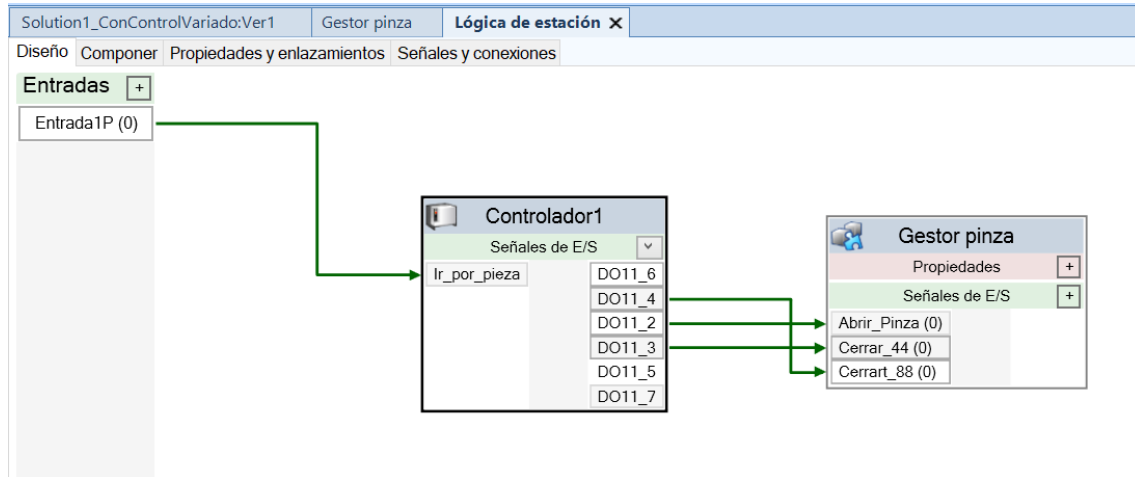


Figura 4. 106. Diseño de la lógica de estación

Para llevar a cabo estas acciones se ha incluido dentro del gestor de la pinza las siguientes propiedades [Figura 107]:

- Cuando se produce la salida correspondiente a cerrar la pinza a 44 mm, la herramienta pinza va hasta la posición correspondiente y la pieza de 44 mm queda unida al movimiento que realizará el robot (esta unión desaparecerá junto con la orden abrir pinza).
- Cuando se produce la salida correspondiente a cerrar la pinza a 88 mm, la herramienta pinza va hasta la posición correspondiente y la pieza de 88 mm queda unida al movimiento que realizará el robot (esta unión desaparecerá junto con la orden abrir pinza).
- Cuando se produce la orden abrir pinza, el mecanismo correspondiente a la herramienta pinza realiza el movimiento de apertura. Tras ello se desanclarán tanto la pieza de 44 mm como la de 88 mm del movimiento del robot, ya que al abrir la pinza se han soltado la pieza que se manipulaba.

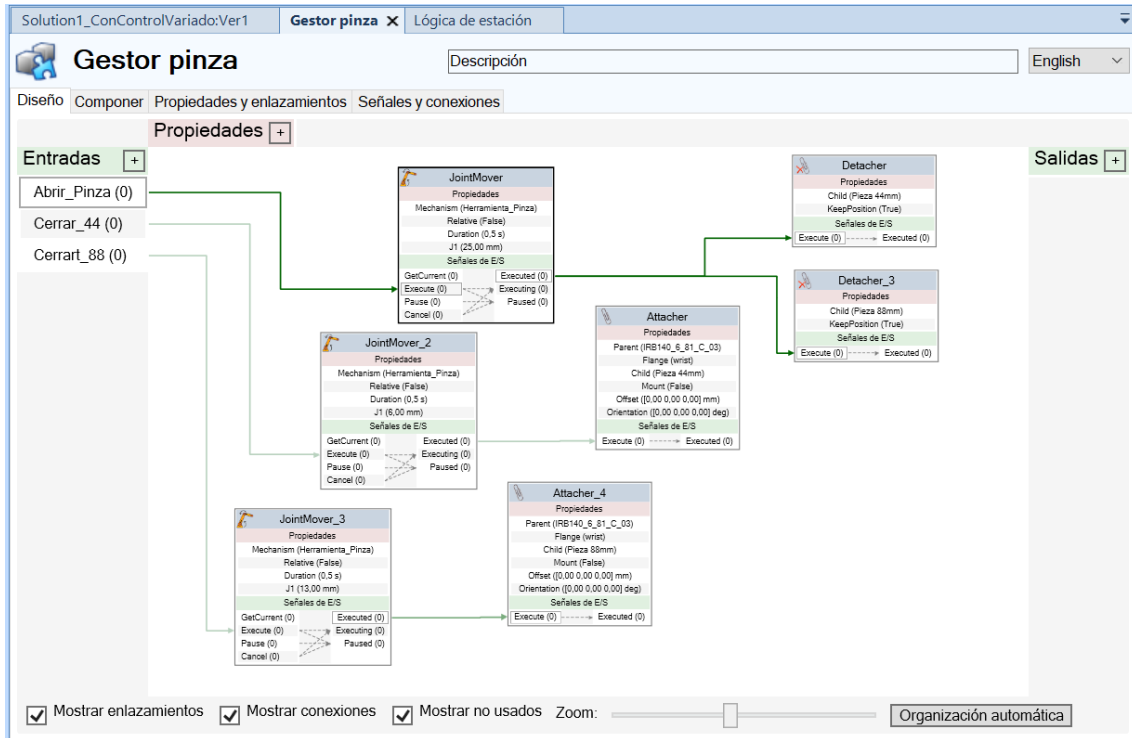


Figura 4. 107. Lógica del gestor de la pinza

4.6.5. PROGRAMACION DE LAS ACCIONES MEDIANTE LENGUAJE RAPID

Una vez configurado todo lo necesario, se procederá a la programación mediante lenguaje Rapid de las acciones a realizar por el robot.

Esta programación, a través de las salidas que se generarán en la tarjeta DSQC 653, tendrán sus correspondientes repercusiones en forma de acciones en la herramienta, como se ha explicado en el apartado dedicado a la programación mediante Arduino.

Las fases de las que se compone nuestro código son las siguientes.

1. **Puntos del programa:** En primer lugar, pueden observarse todos los puntos definidos que se han descrito anteriormente.

```

1  MODULE Module1
2  CONST robtarget Target_10:=[[0,-100,100],[0,0,1,0],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
3  CONST robtarget Target_20:=[[0,-80,125,-30],[0.5,0.5,0.5,-0.5],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
4  CONST robtarget Target_30:=[[0,300,-110],[0.5,0.5,0.5,-0.5],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
5  CONST robtarget Target_40:=[[80,125,-30],[0.5,0.5,0.5,-0.5],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
6  CONST robtarget Target_50:=[[444,105,-110],[0.5,0.5,0.5,-0.5],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
7  CONST robtarget Target_60:=[[444,105,-110],[0.5,0.5,0.5,-0.5],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
8  CONST robtarget Home:=[[662.176223927,0,589.5],[0.5,0,0.866025404,0],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
9  CONST robtarget Target_80:=[[400,400,-110],[0.5,0,0.5,-0.707106781],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
10 CONST robtarget Target_70:=[[400,400,-110],[0.707106781,0.5,0,-0.5],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
11 CONST robtarget Inicio_10:=[[662.176223927,0,589.5],[0.5,0,0.866025404,0],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
12 CONST robtarget Target_90:=[[80,175,-190],[0.5,0.5,0.5,-0.5],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
13 CONST robtarget Target_100:=[[80,175,-190],[0.5,0.5,0.5,-0.5],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];

```

2. Datos del programa: nombre, autor, etc.

```
14 |  
15 |  
16 | !*****  
17 | ! Módulo: Module1  
18 | ! Descripción:  
19 | ! Programa para la prueba de trayectoria del robot usando pinza  
20 | ! Autor: Borja Hierro  
21 | ! Versión: 1.0  
22 | !*****  
23 |  
24 |  
25 |  
26 |
```

3. Trayectorias creadas manualmente: Al sincronizar la estación de RobotStudio con el lenguaje Rapid, se han creado las trayectorias definidas manualmente:

```
27 | PROC INICIO()  
28 |     MoveL Inicio_10,v100,z100,TCPPinza_Ventosa\WObj:=wobj0;  
29 | ENDPROC  
30 |  
31 | PROC Ir_a_coger_disco()  
32 |     MoveJ Inicio_10,v100,z100,TCPPinza_Ventosa\WObj:=wobj0;  
33 |     MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;  
34 |     MoveJ Target_40,v100,fine,TCPPinza_Ventosa\WObj:=wobj_discos;  
35 | ENDPROC  
36 | PROC Ir_a_dejar_disco()  
37 |     MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;  
38 |     MoveJ Target_20,v100,fine,TCPPinza_Ventosa\WObj:=wobj_discos;  
39 | ENDPROC  
40 | PROC Ir_a_coger_pieza_grande()  
41 |     MoveJ Target_80,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;  
42 |     MoveJ Target_50,v100,fine,TCPPinza_Ventosa\WObj:=wobj_discos;  
43 | ENDPROC  
44 |  
45 | PROC Ir_a_dejar_pieza_grande()  
46 |     MoveJ Target_70,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;  
47 |     MoveJ Target_60,v100,fine,TCPPinza_Ventosa\WObj:=wobj_discos;  
48 | ENDPROC  
49 |  
50 | PROC Ir_a_coger_pequeño()  
51 |     MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;  
52 |     MoveJ Target_90,v100,fine,TCPPinza_Ventosa\WObj:=wobj_discos;  
53 | ENDPROC  
54 |  
55 | PROC Ir_a_dejar_pequeño()  
56 |     MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;  
57 |     MoveJ Target_100,v100,fine,TCPPinza_Ventosa\WObj:=wobj_discos;  
58 | ENDPROC
```

4. **Programa principal:** esta parte del programa recibirá el nombre de main, y en él se encuentra el código que ejecutará el controlador.

Nuestro programa se dividirá en las siguientes secuencias:

- En primer lugar, se reinician y ponen a 0 todas las salidas del módulo DSQC653 que estará conectado a la siguiente fase de control. De manera que la herramienta pinza no reciba ninguna orden.

```
60      !*****
61      ! Procedimiento Main
62      !*****
63
64      PROC main()
65          !WaitDI Armar,1;
66          INICIO;
67          !Reseteo todas las salidas
68          Reset D011_2;
69          Reset D011_3;
70          Reset D011_4;
71          Reset D011_5;
72          Reset D011_6;
73          Reset D011_7;
74
```

- Tras ello, se activará la salida que abre la pinza, se esperará un segundo y se desactivará. De esta manera la pinza del robot quedará abierta pero no se estará ocupando el control de esta con la orden de abrir.

```
74
75      !Set y reset de D011_2 para abrir la pinza
76      set D011_2;
77      WaitTime 1;
78      Reset D011_2;
79
```

Una vez realizado este proceso, se programan las distintas acciones de robot:

1ª Acción: mover la pieza grande:

Se realizarán los siguientes pasos:

1. Con la trayectoria denominada *lr_a_coger_pieza_grande* el robot se posicionará sobre la pieza grande.

2. Tras ello, esperará 2 segundos y se activará la salida correspondiente a cerrar la pinza a una distancia de 88 mm correspondiente al tamaño de esta pieza.
3. Una vez la pinza ha cogido la pieza, se espera 3 segundos y se la llevará a hasta la posición de inicio pasando por puntos que aseguran la no colisión del robot.
4. De manera continua a este último movimiento el robot se dirige al punto donde ha de dejarla
5. Tras esperar 2 segundos reinicia la salida que cierra la pinza a 88 mm y activa la salida que abre la pinza, consiguiendo así soltar la pieza en su nueva posición.
6. Tras esperar 3 segundos más, el robot volverá a su posición de inicio.

```
80      !Ciclo coger pieza grande (pinza cierra a 88mm)
81
82      Ir_a_coger_pieza_grande;
83      WaitTime 2;
84      Set DO11_4; !Activo la salida de cerrar pinza a 88mm
85      WaitTime 3;
86      MoveJ Target_80,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;
87      INICIO;
88
89      !Pieza grande cogida, voy a dejarla
90      Ir_a_dejar_pieza_grande;
91      WaitTime 2;
92      Reset DO11_4;!Pongo a 0 la salida de cerrar pinza a 88mm
93      Set DO11_2; !Activo la salida que abre la pinza
94      WaitTime 3;
95      MoveJ Target_70,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;
96      INICIO;
97      Reset DO11_2; !reseteo la salida de pinza abierta
```

2ª Acción: mover el disco

Partiendo de la posición inicio en la que se encuentra el robot tras finalizar su acción anterior, el robot realizará las siguientes acciones:

1. Trayectoria hasta el punto de coger el disco.
2. Espera 1 segundo.
3. Activa la salida que acciona la succión de la ventosa.
4. Espera 4 segundos y la desactiva.
5. En el instante de desactivar la succión, activa la salida correspondiente a mantener el vacío, de manera que el disco queda sujeto a la herramienta.
6. Traslado el disco hasta su nueva posición.
7. Se desactiva la variable que mantiene el vacío y tras ello se activa la variable que quita la presión en las ventosas, consiguiendo así dejar la pieza.
8. Espera 3 segundos y vuelve a la posición de inicio tras reiniciar la anterior variable.

```
99      !Ciclo coger discos
100
101      Ir_a_coger_disco;
102      WaitTime 1;
103      Set D011_5; !Activo la salida de succion con la ventosa
104      WaitTime 4; !4 segundos de succion
105      reSet D011_5; !Desactivo la succion
106      Set D011_6; !Activo la variable que mantiene el vacio y la dejo
107      MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;
108
109      !Pieza cogida con vacio, voy a dejarla
110      Ir_a_dejar_disco;
111      WaitTime 2;
112      Reset D011_6;!La ventosa suelta el vacio
113      Set D011_7;!La válvula suelta la pieza
114      WaitTime 3;
115      Reset D011_7;
116      MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;
117      INICIO;
```

3ª Acción: mover la pieza grande

Esta acción tendrá un desarrollo igual que la primera acción referente a mover la pieza grande, salvo que en ella las variables activadas estarán referidas al cierre de la pinza para coger una pieza de 44 mm de ancho y los puntos a los que se desplaza el robot son los correspondientes a la pieza pequeña.

```
119      !Ciclo coger pieza pequeña (pinza cierra a 44mm)
120
121      Ir_a_coger_pequeño;
122      WaitTime 1;
123      Set D011_3; !Activo la salida de cerrar pinza a 44mm
124      WaitTime 4;
125      MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;
126
127      !Pieza pequeña cogida, voy a dejarla
128      Ir_a_dejar_pequeño;
129      WaitTime 2;
130      Reset D011_3;!Pongo a 0 la salida de cerrar pinza a 88mm
131      Set D011_2; !Activo la salida que abre la pinza
132      WaitTime 3;
133      MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;
134      INICIO;
135      Reset D011_2; !reseteo la salida de pinza abierta
136
137      !Discos;
138
139      ENDPROC
```

Cabe destacar la aplicación de la aproximación tipo *fine* a los puntos donde la herramienta entra en contacto con las piezas a manipular, ya que, de esta manera, la herramienta llegará de forma precisa al punto exacto marcado en el programa.

De igual manera, se ha bajado la velocidad de desplazamiento para una simulación más clara y para una mayor seguridad a la hora de realizar la puesta en marcha real del sistema.

4.6.6. SIMULACIÓN DEL SISTEMA

Una vez se sincroniza la programación realizada mediante Rapid con la estación, se podrán simular todos los movimientos y acciones que tienen lugar en nuestro programa.

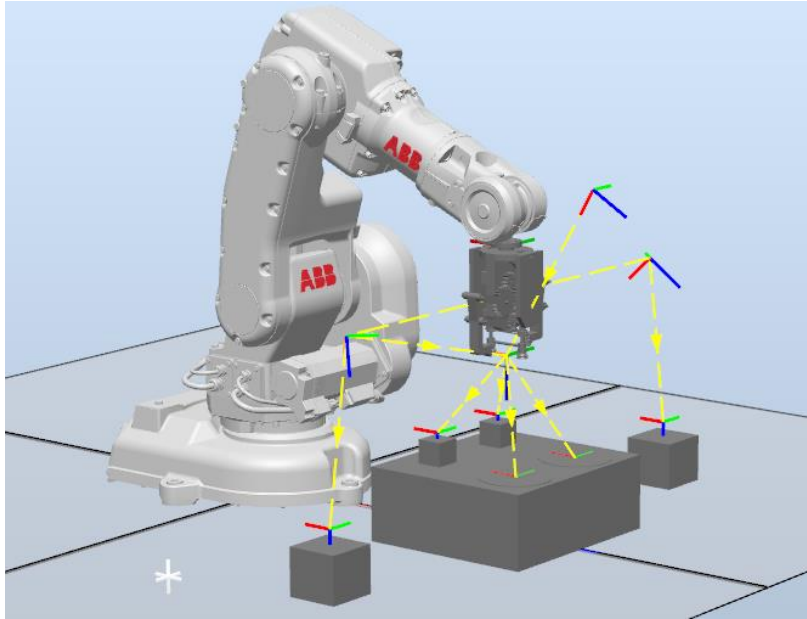


Figura 4. 108. Visualización de trayectorias simuladas

Se podrá observar también los estados de las salidas mediante el simulador de entradas y salidas que contiene Robot Studio.

La simulación llevada a cabo se puede ver a través de los siguientes enlaces:

- Video 1: Pantalla robotstudio en el que se puede observar la evolución de las salidas además de las trayectorias del programa:

<https://www.youtube.com/watch?v=4ZKIYF1n2fU>

- Video 2: Simulación en pantalla completa sin entradas/salidas:

<https://youtu.be/YRwdYCF6qnM>

Tras el proceso de simulación, se podrá concluir que la programación llevada a cabo para el control del conjunto robot-herramienta a través de RobotStudio es satisfactorio.

4.7. PUESTA EN MARCHA DEL SISTEMA REAL (SISTEMA ROBOT-HERRAMIENTA)

Una vez se ha fabricado, ensamblado y programado todo el conjunto que engloba este proyecto, tan solo faltará verificar su correcto funcionamiento mediante una prueba real en las instalaciones de la Universidad de Almería.

Para llevar a cabo este proceso de prueba se han seguido los siguientes puntos:

1. En primer lugar, se han transferido los archivos desde el programa RobotStudio al controlador del robot y se ha comprobado que el robot reproduzca correctamente la trayectoria programada. De igual manera se ha comprobado que a través de la programación se obtienen las señales de salida correctamente en el módulo DSQC 653. Esto es posible gracias a que este módulo cuenta con señales luminosas para cada una de sus salidas. Se realizan estas comprobaciones y se verifica que los resultados obtenidos son satisfactorios ya que el robot alcanza correctamente los puntos donde estarán las piezas a manipular.

2. El segundo paso por realizar es el acoplamiento de la herramienta al brazo robótico. Este proceso ha sido mostrado con anterioridad en el punto 4.4 *HERRAMIENTA FINAL COMPLETA*. Se ha realizado adicionalmente una prueba para comprobar el correcto funcionamiento del sistema encargado de medir y mostrar la distancia:

- Video 3: Sensor de distancia:

<https://youtu.be/pib2c2WYlt4>

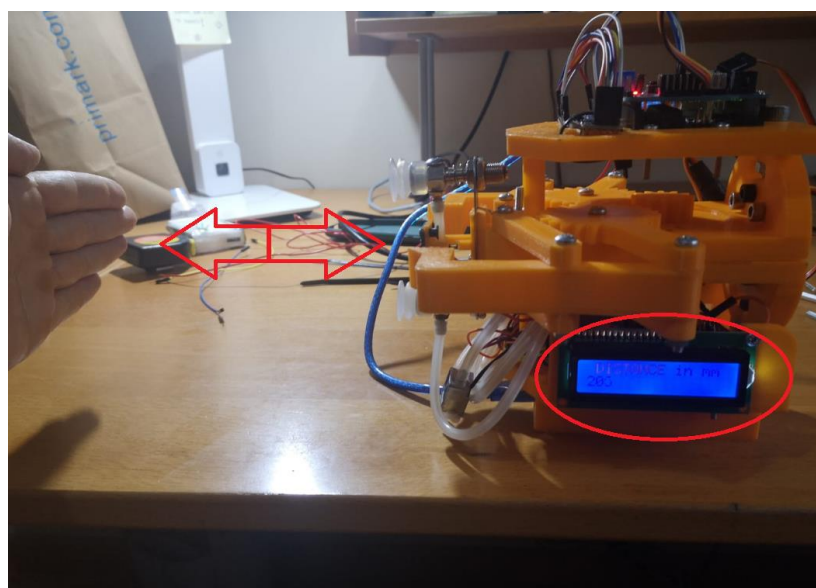


Figura 4. 109. Funcionamiento sensor distancia

Una vez comprobado el correcto funcionamiento de los componentes, se ensamblará la herramienta con el robot tal y como se puede observar en la figura 4.111.

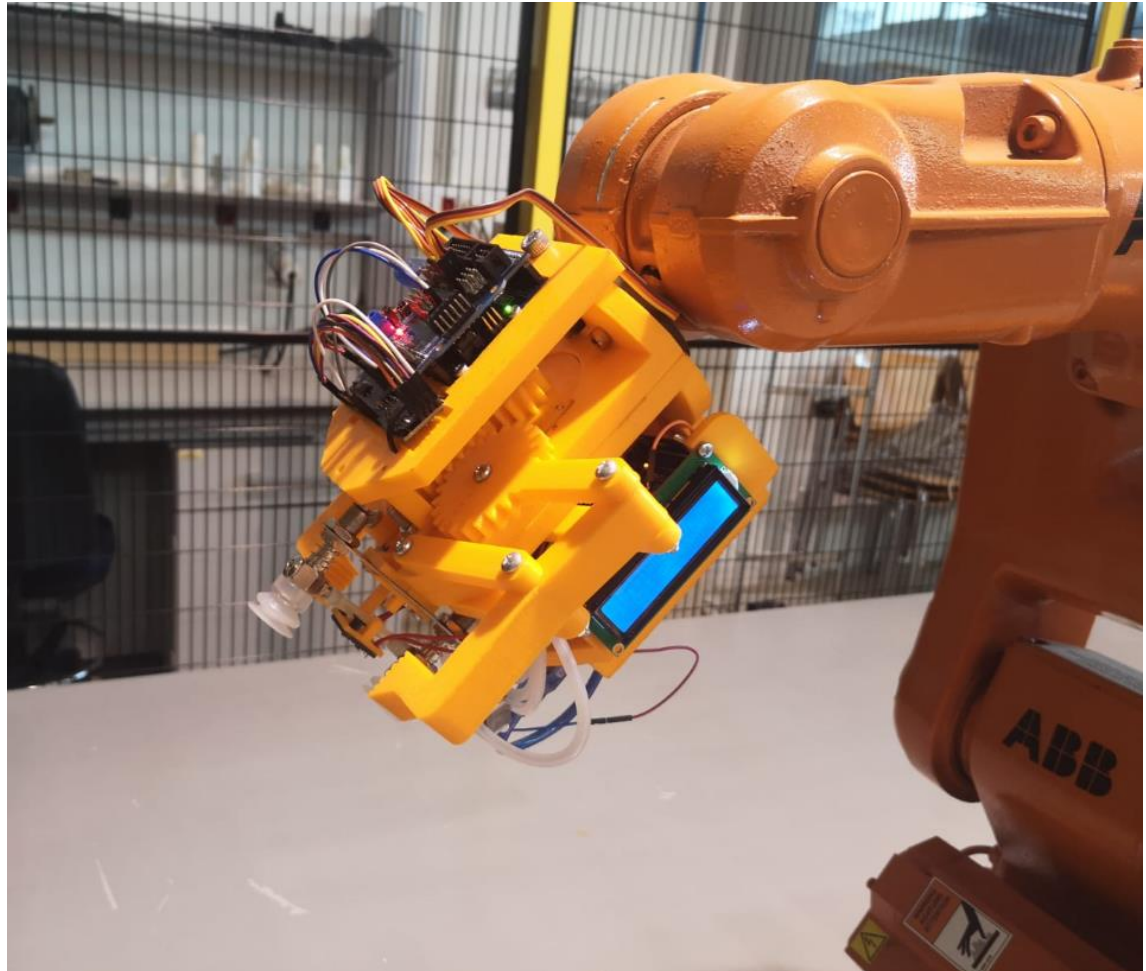


Figura 4. 110. Ensamble herramienta-robot

3. Tras el acoplamiento de la herramienta al sistema del robot se ha procedido a la realización de una prueba de funcionamiento en el modo manual. En ella, el robot será controlado a través de su controlador IRC5C, mientras que la herramienta pinza-ventosa se ha controlado a través de la ProtoBoard.

El resultado obtenido es totalmente satisfactorio, puesto que el conjunto realiza la tarea para la que se ha diseñado a la perfección, como se puede observar en el video del siguiente enlace.

- Video 4: control ProtoBoard:

<https://youtu.be/BzJU6cxk58>

4. Una vez que se ha comprobado que el control manual de la herramienta funciona correctamente, se procede a realizar el cableado de la ProtoBoard con el módulo de salidas del controlador del robot. Este montaje se realiza a través de cableado de prueba [Figura 4.112].

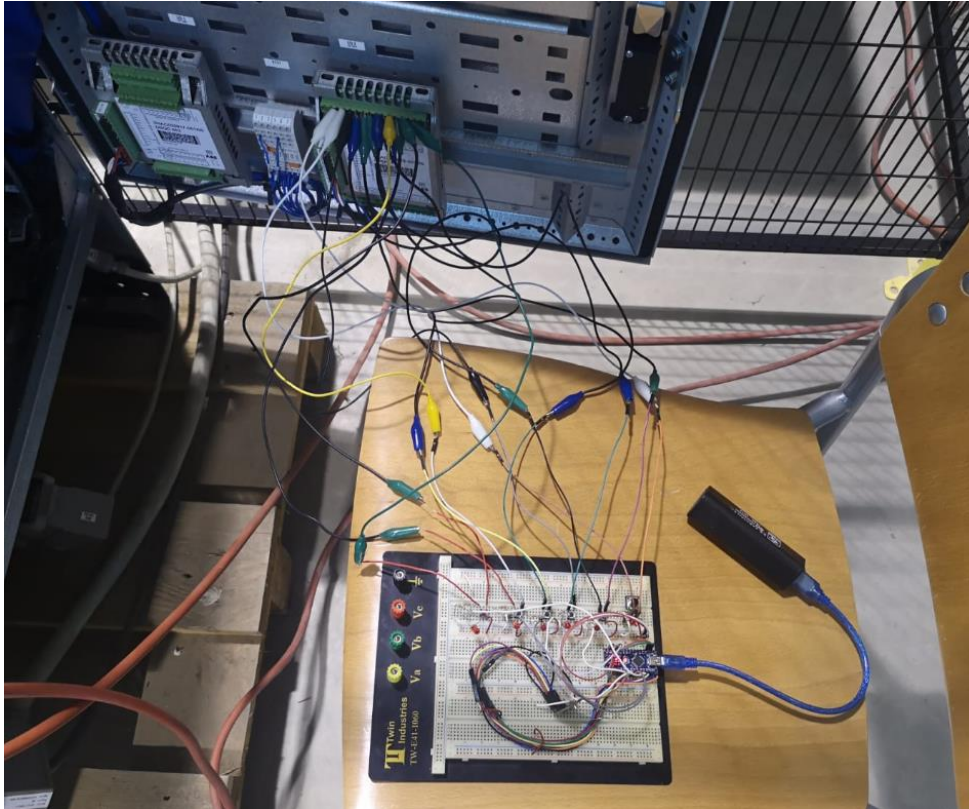


Figura 4. 111. Cableado para unión de los sistemas

El resultado es satisfactorio ya que se observa que, al activarse las salidas del módulo DSQC653, el led correspondiente de la ProtoBoard también se iluminará, lo cual indica que los distintos pines del Arduino Nano de la ProtoBoard se estarán activando correctamente mediante las salidas que proporciona el IRC5C. Este cableado puede observarse en la figura 4.112 y en los primeros instantes del **video 2** que se muestra en el siguiente punto.

Cabe mencionar que el sistema que proporciona la alimentación necesaria para el funcionamiento de los sistemas incluidos en la ProtoBoard y el módulo de salidas será una PowerBank que alimenta directamente al Arduino Nano y de ahí transmite el voltaje correspondiente para alimentar los demás sistemas.

5. Una vez se ha realizado el montaje completo del sistema, se procede a su prueba a través del control autónomo diseñado para el proyecto. Esta prueba se refleja dentro de los videos que se muestran a continuación:
- Video 5: prueba control automatizado:

<https://youtu.be/Bnm0Ed5en8w>

Se ha realizado adicionalmente un video con especial atención a la herramienta pinza-ventosa en el que se observa su funcionamiento.

- Video 6: detalle de funcionamiento:

<https://youtu.be/pIVPgH1JU0>

Como puede observarse dentro de los videos incluidos en este apartado, el resultado de la implementación real de nuestro sistema es completamente satisfactorio, puesto que cumple con todos los requisitos que se establecieron dentro del proyecto.

4.8. MEJORAS Y TRABAJOS FUTUROS

Dentro de las posibles mejoras y propuestas para trabajos futuros relacionados con este TFM, se incluirán:

- La posibilidad de un sistema de control más compacto, reduciendo el número de componentes que se utilizan dentro de este diseño.
Partiendo de posibilidades menos definidas dentro de los componentes de control disponibles, se podrá buscar una solución que permita reducir el número de componentes y de esa forma proporcionar a la herramienta un aspecto más compacto.
- Adición de sensores que aporten más información a la herramienta, como lo será un sensor de giro que indique la orientación de la herramienta en todo momento.
- Adición de un mayor número de programas con mayor complejidad para realizar distintas tareas haciendo uso de la configuración actual de la herramienta.



5. CONCLUSIONES

Al comienzo de esta memoria se establecieron unos objetivos y subobjetivos que se han conseguido cumplir satisfactoriamente.

En primer lugar, se han conseguido diseñar y verificar todos los elementos mecánicos necesarios de la herramienta de manera que se ajusten a las especificaciones iniciales. Estos mismos elementos diseñados se han impreso mediante tecnología 3D obteniendo unos resultados satisfactorios.

Tras ello, se han comprobado tanto virtual como físicamente el correcto ensamble entre los componentes de la herramienta y sus sistemas acoplados.

Se ha alcanzado también el objetivo de dotar al sistema de un control autónomo a través de programación de los distintos elementos. Adicionalmente se ha incluido la posibilidad de control manual de la herramienta, aportando así flexibilidad al sistema.

Por último, se han realizados las pruebas correspondientes para verificar que la puesta en marcha del sistema real es correcta.

Con todo ello se podrá concluir que el resultado de este proyecto cumple con todos los requisitos establecidos y es satisfactorio.



Diseño y fabricación de una herramienta multifunción para un robot industrial.

6. REVISIÓN BIBLIOGRÁFICA

- [1] NORTON, Robert L. Diseño de maquinaria: una introducción a la síntesis y al análisis de mecanismos y maquinas. 1995.
- [2] BUDYNAS, Richard G.; NISBETT, J. Keith. Diseño en ingeniería mecánica de Shigley. McGraw-Hill Interamericana, 2012.
- [3] BARRIENTOS, Antonio, et al. Fundamentos de robótica. Madrid: McGraw-Hill, 2007.
- [4] GROOVER, Mikell P. Fundamentos de manufactura moderna: materiales, procesos y sistemas. Pearson Educación, 1997.
- [5] DOMÍNGUEZ ABASCAL, Jaime. Teoría de máquinas y mecanismos. Universidad de Sevilla, 2016.
- [6] LLAMAS, Luis. Tutoriales de Arduino. Acceso el 16 de febrero de 2021.
<https://www.luisllamas.es/tutoriales-de-arduino/>
- [7] Arduino Language Reference. Acceso el 15 de febrero de 2021.
<https://www.arduino.cc/reference/en/>
- [8] Manual del operador RobotStudio. Acceso el 5 de marzo de 2021.
https://library.e.abb.com/public/6aeb483836740e11c1257b4b0052375b/3HAC032104-005_revE_es.pdf
- [9] Impresión 3D: qué es, métodos, aplicaciones, materiales e impresoras 3D. Acceso 10 de mayo 2021.
<https://www.adslzone.net/reportajes/tecnologia/impresion-3d/>
- [10] Estructura de un robot industrial. Acceso 13 de mayo 2021.
http://platea.pntic.mec.es/vgonzale/cyr_0204/ctrl_rob/robotica/sistema/morfologia.htm



Diseño y fabricación de una herramienta multifunción para un robot industrial.

ANEXO 1: CÁLCULOS

1. SÍNTESIS DE NÚMERO DE UN MECANISMO DE 4 BARRAS

En primer lugar, se realiza la comprobación de que el mecanismo adoptado para el diseño de este problema tan solo tenga 1 grado de libertad. [5] Esto se realiza puesto que es necesario que, para un ángulo dado de movimiento del eslabón conectado al servomotor, el mecanismo tenga una posición única y establecida.

$$G = 3(N - 1) - 2C_I - C_{II}$$

G : grado de libertad

N : número de eslabones

C_I : número de juntas de 1gdl (completas)

C_{II} : número de juntas de 2gdl (semi)

El mecanismo de 4 barras de nuestro problema es el siguiente [Figura Anexos 1]:

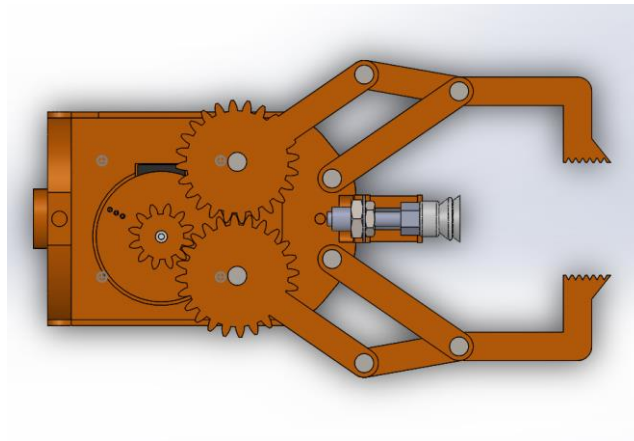


Figura Anexos 1. Mecanismo de 4 barras implementado

Podemos ver el mecanismo de manera clara en los eslabones que forman la pinza [Figura Anexos 2]:

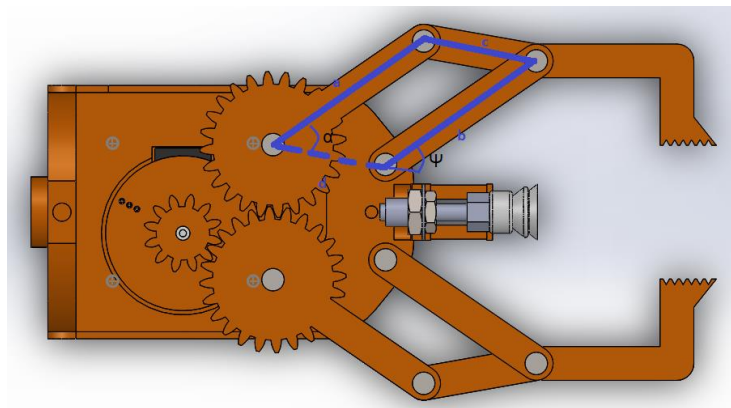


Figura Anexos 2. Geometría del mecanismo de 4 barras implementado

Cabe destacar que la pieza que forma el eslabón c actuará a efectos del mecanismo como la barra indicada en la figura.

Simplificando nuestro mecanismo de 4 barras para un análisis claro obtenemos un mecanismo tipo [Figura Anexos 3]:

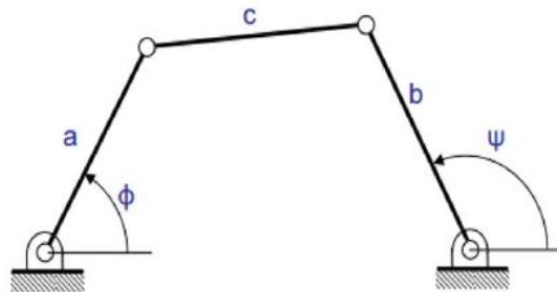


Figura Anexos 3. Elementos del mecanismo de 4 barras genérico

Dentro de nuestro problema tendremos

- $C_I = 4$ juntas de 1gdl:
 - Binarios (junta de 2 barras): 4
 - Ternarios (junta de 3 barras): 0
- $C_{II} = 0$ juntas de 2gdl:
- $N = 4$ barras

$$G = 3(4 - 1) - 2 \cdot 4 - 0;$$

$$G = 1;$$

Para realizar la comprobación de que se cumple este requisito dentro de nuestro mecanismo de 4 barras, hemos de estudiar el cumplimiento de la ecuación modificada de síntesis de número de Grubler:

$$N - (G + 3) + 2t + 6q + \dots = T + 2Q + \dots$$

G : grado de libertad = 1

N : número de eslabones = 4

t : pares ternarios = 0

q : pares cuaternarios = 0

T : número de eslabones ternarios = 0

Q : eslabones cuaternarios = 0

Solo hay pares binarios, es por ello por lo que t y q son 0. Pasa igual con los eslabones, al solo haber eslabones binarios T y Q serán 0.

Comprobamos el cumplimiento de la ecuación:

$$4 - (1 + 3) + 0 + 0 = 0 + 0; \text{ Se cumple.}$$

2. SÍNTESIS DIMENSIONAL DEL MECANISMO DE LA PINZA

Para conseguir el paralelismo entre los lados opuestos del mecanismo de 4 barras (a paralelo a b y c paralelo a d), es necesario que los ángulos de giro φ y Ψ sean iguales. [Figura Anexos 4]

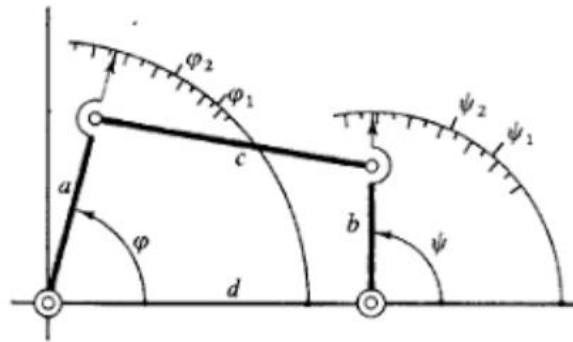


Figura Anexos 4. Ángulos dentro del mecanismo de 4 barras

A través de la ecuación de Freudenstein para 2 puntos de precisión se demuestra que los lados del mecanismo a diseñar han de ser iguales 2 a 2 para cumplir este paralelismo buscado.

Ecuación de Freudenstein:

$$K_1 \cos \Psi_1 - K_2 \cos \varphi_1 + K_3 = \cos (\varphi_1 - \Psi_1)$$

Donde:

$$K_1 = \frac{d}{a}$$

$$K_2 = \frac{d}{b}$$

$$K_3 = \frac{d^2 + b^2 + a^2 - c^2}{2ab}$$

Condición que quiero que se cumpla: $\varphi_1 = \Psi_1$

Contando con ello, la ecuación de Freudenstein quedará:

$$K_1 \cos \varphi_1 - K_2 \cos \varphi_1 + K_3 = \cos 0;$$

$$(K_1 - K_2) \cos \varphi_1 + K_3 = 1$$

Aplicando esta condición a los dos ángulos extremos que se desean conseguir con el movimiento de mi mecanismo de 4 barras (0° para el mecanismo cerrado; 90° para el mecanismo totalmente abierto) puedo obtener las condiciones buscadas:

Para $\varphi_1 = 90^\circ$:

$$(K_1 - K_2) \cos 90 + K_3 = 1;$$

$$(K_1 - K_2) \cdot 0 + K_3 = 1;$$

$$K_3 = 1$$

Para $\varphi_2 = 0^\circ$:

$$(K_1 - K_2) \cos 0 + 1 = 1;$$

$$(K_1 - K_2) \cdot 1 = 1 - 1;$$

$$K_1 - K_2 = 0;$$

$$K_1 = K_2$$

Conociendo esto y aplicando la fórmula de K1 y K2 respectivamente:

$$K_1 = K_2;$$

$$\frac{d}{a} = \frac{d}{b};$$

$$a = b$$

Conociendo que a es igual a b, también podremos obtener la relación entre c y d a través del coeficiente K3, que previamente conocíamos que es igual a 1:

$$K_3 = \frac{d^2 + b^2 + a^2 - c^2}{2ab};$$

$$K_3 = \frac{d^2 + a^2 + a^2 - c^2}{2a^2};$$

$$K_3 = \frac{d^2 - c^2 + 2a^2}{2a^2};$$

$$1 = \frac{d^2 - c^2}{2a^2} + 1;$$

$$0 = \frac{d^2 - c^2}{2a^2};$$

$$0 = d^2 - c^2;$$

$$c = d$$

Con ello queda demostrado que para conseguir los ángulos φ y Ψ sean iguales, será condición obligatoria que las distancias de los lados a y b sean iguales, así como la distancia de c y d. Con ello se conseguirá mantener siempre el paralelismo entre los lados opuestos del mecanismo.

3. DIMENSIONAMIENTO DEL SISTEMA DE ENGRANAJES

En primer lugar, han de fijarse una serie de parámetros de diseño dentro del sistema de engranajes.

Para la selección del módulo se ha tenido en cuenta que la utilización de tamaños de módulo pequeños planteará probablemente un número de dientes excesivo en la rueda. En el otro extremo, si el tamaño del módulo es grande puede aparecer el problema de interferencia en el piñón. Es por ello por lo que se selecciona un módulo $m=2$.

Para el diámetro de paso del engranaje de entrada se ha tenido en cuenta que este debe de quedar dentro de la base de la herramienta, por lo cual se ha determinado un diámetro de 24 mm para que no exista problema a la hora de incluirlo en el diseño.

Parámetros de diseño fijados:

- Relación de transmisión = $\frac{1}{2}$
- Diámetro de paso engranaje de entrada: $D_E = 24$ mm
- Módulo del engranaje de entrada $m_E = 2$

Con ello se podrá calcular los parámetros de nuestro engranaje de salida:

Módulos de engranajes que funcionan conjuntamente es el mismo: $m_S = 2$

$$i = \frac{1}{2};$$
$$i = \frac{D_E}{D_S} = \frac{z_E}{z_S};$$

Obtenemos en primer lugar el diámetro de paso del engranaje de salida:

$$\frac{D_E}{D_S} = \frac{1}{2};$$

$$D_S = 2 \cdot 24 = 48 \text{ mm};$$

Obtenemos también su número de dientes:

$$z_S = \frac{D_S}{m} = \frac{48}{2} = 24 \text{ dientes}$$

Siendo también el número de dientes del eslabón de entrada:

$$z_E = \frac{D_E}{m} = \frac{24}{2} = 12 \text{ dientes}$$

La tabla resumen correspondiente es la que podemos encontrar a continuación [Tabla 24].

	ESLABON DE ENTRADA	ESLABONES DE SALIDA
DIAMETRO DE PASO	24	48
NUMERO DE DIENTES	12	24
MODULO	2	2

Tabla 24. Configuración de los eslabones

4. PUNTOS DE CIERRE PARA LA MANIPULACIÓN DE LAS DISTINTAS PIEZAS

En este apartado se calculará el ángulo necesario que se ha de dar al servomotor encargado del cierre de las pinzas para manipular correctamente los 2 tipos distintos de piezas.

Manipulación de pieza grande:

- Dimensiones [Figura Anexos 5]:
 - Altura: 40 mm
 - Anchura: 88 mm
 - Gosor: 88 mm

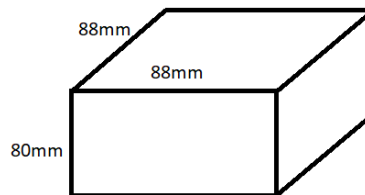


Figura Anexos 5. Dimensiones pieza grande

Se calcularán los grados de cierre necesario para conseguir una apertura de la pinza de 88 mm.

Para ello, mediante método gráfico se obtiene que el ángulo existente cuando el cierre de la pinza es total es de 11,58º, que corresponderá a la instrucción 0º dentro de la programación de nuestro servomotor [Figura Anexos 6].

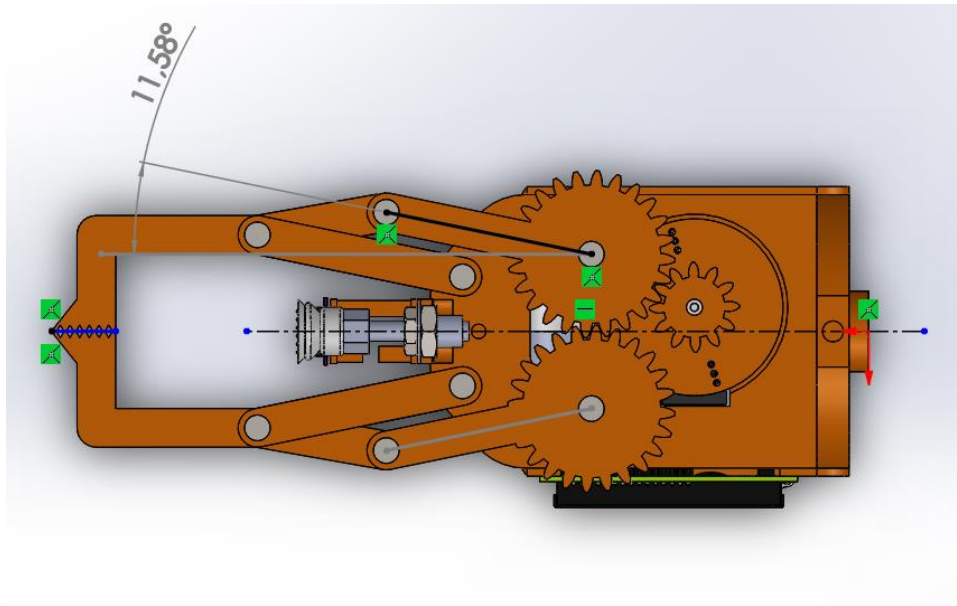


Figura Anexos 6. Ángulo en cierre máximo

Se calcula cual es el ángulo existente cuando la apertura es de 88 mm [Figura Anexos 7]:

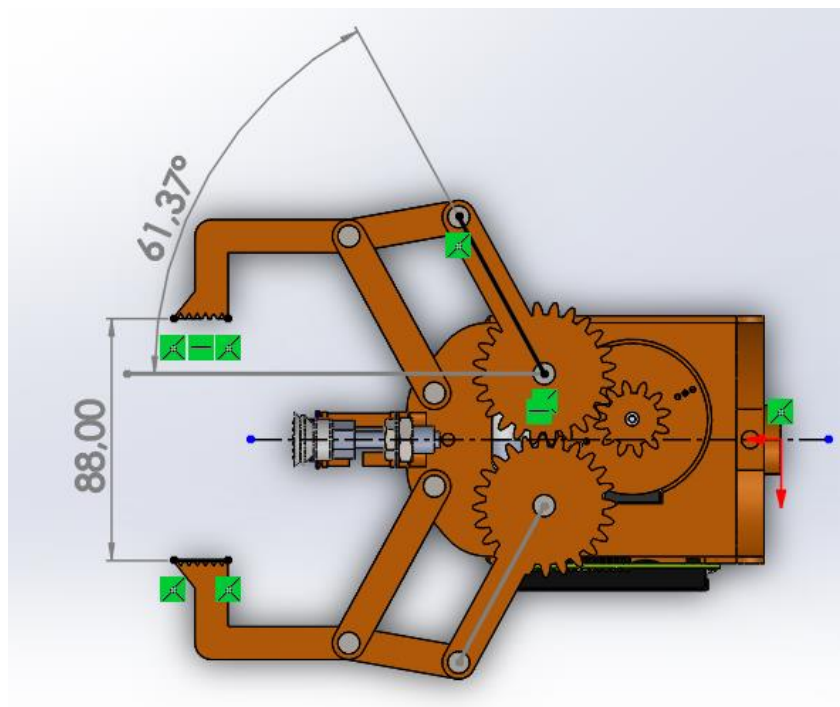


Figura Anexos 7. Ángulos en apertura de 88 mm

El ángulo existente es de 61,37°.

El aumento de la apertura dentro de nuestro sistema de pinzas es de $49,79^\circ$. Sabiendo que la relación de transmisión entre el engranaje del eslabón de la pinza y el engranaje del servomotor es de 2 a 1. El ángulo que ha de girar el servomotor partiendo de la posición 0° es el doble de lo que se ha incrementado la apertura del eslabón.

$$\alpha = 2 \cdot 49,79 = 99,58^\circ$$

El ángulo para conseguir la apertura de 88 mm es de $99,58^\circ$.

Manipulación de pieza pequeña [Figura Anexos 8]:

- Dimensiones:
 - Altura: 40 mm
 - Anchura: 44 mm
 - Grosor: 44 mm

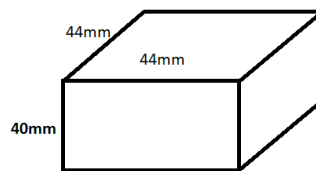


Figura Anexos 8. Pieza pequeña 44 mm

Ya conocemos que el ángulo de cierre total es de $11,58^\circ$. Ahora se calcula cual es el ángulo existente cuando la apertura es de 44 mm [Figura Anexos 9]:

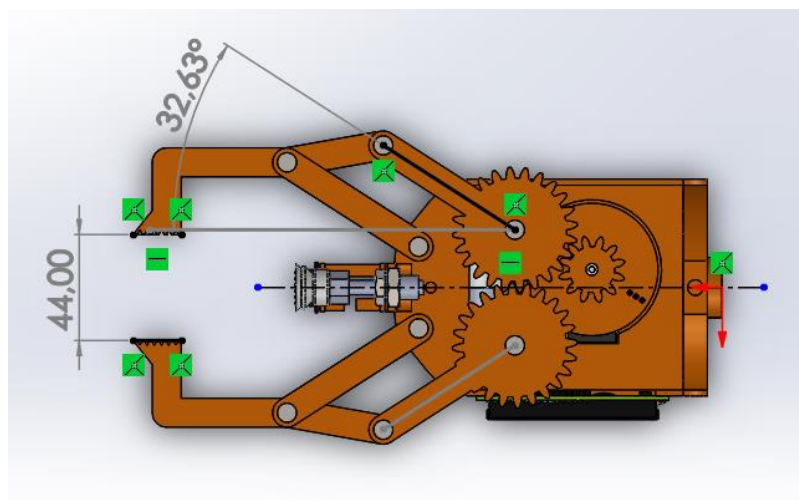


Figura Anexos 9 Ángulo en apertura de 44 mm

El ángulo existente es de $32,63^\circ$.

El aumento de la apertura dentro de nuestro sistema de pinzas es de $21,05^\circ$. Sabiendo que la relación de transmisión entre el engranaje del eslabón de la pinza y el engranaje del servomotor es de 2 a 1. El ángulo que ha de girar el servomotor

partiendo de la posición 0º es el doble de lo que se ha incrementado la apertura del eslabón.

$$\alpha = 2 \cdot 21,05 = 42,1^\circ$$

El ángulo para conseguir la apertura de 44 mm es de 42,1º.

5. ANÁLISIS DE LAS PIEZAS MEDIANTE SOLIDWORKS SIMULATION

5.1. SOPORTE SUPERIOR

- Análisis realizado: análisis estático.
- Sujeciones: anclaje mediante pernos a la base.
- Fuerzas aplicadas debido a los pesos de sus componentes:
 - Arduino uno → 0,2646N
 - Shield v5.0 → 0,245N
 - Arduino nano + módulo de radio → 0,1274N

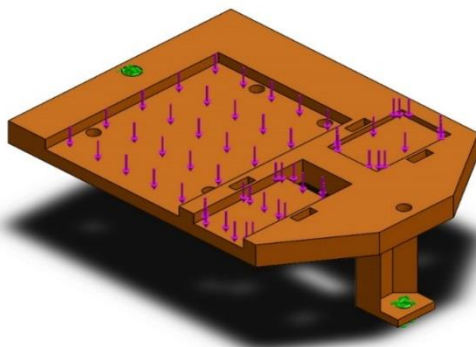


Figura Anexos 10. Fuerzas aplicadas sobre el soporte superior.

Nombre de carga	Cargar imagen	Detalles de carga
Fuerza-1		Entidades: 1 cara(s) Tipo: Aplicar fuerza normal Valor: 0,5096 N
Fuerza-2		Entidades: 2 cara(s) Tipo: Aplicar fuerza normal Valor: 0,1274 N

Tabla 25. Fuerzas sobre el soporte superior

- Mallado: tamaño medio

Número total de nodos	15932
Número total de elementos	9083
Cociente máximo de aspecto	10,301
% de elementos cuyo cociente de aspecto es < 3	98,2
% de elementos cuyo cociente de aspecto es > 10	0,011
% de elementos distorsionados (Jacobiana)	0
Tiempo para completar la malla (hh:mm:ss):	00:00:02
Nombre de computadora:	

Nombre del modelo: Soporte
Nombre de estudio: Análisis estático 2(-Predeterminado)
Tipo de malla: Malla sólida

Tabla 26. Mallado soporte superior

RESULTADOS DEL ESTUDIO:

Tensión máxima = $4,5e+04 \text{ N/m}^2 < \text{Límite elástico} = 6,000e+07 \text{ N/m}^2$

Nombre	Tipo	Mín.	Máx.
Tensiones1	VON: Tensión de von Mises	2,223e+01 N/m ² Nodo: 810	4,550e+04 N/m ² Nodo: 15288

Nombre del modelo: Soporte
Nombre de estudio: Análisis estático 2(-Predeterminado)
Tipo de resultado: Análisis estático tensión nodal Tensiones1
Escala de deformación: 6,203,7

Soporte-Análisis estático 2-Tensiones-Tensiones1

Tabla 27. Tensiones soporte superior

- Desplazamientos:

Desplazamiento máximo = 1,959e-03 mm

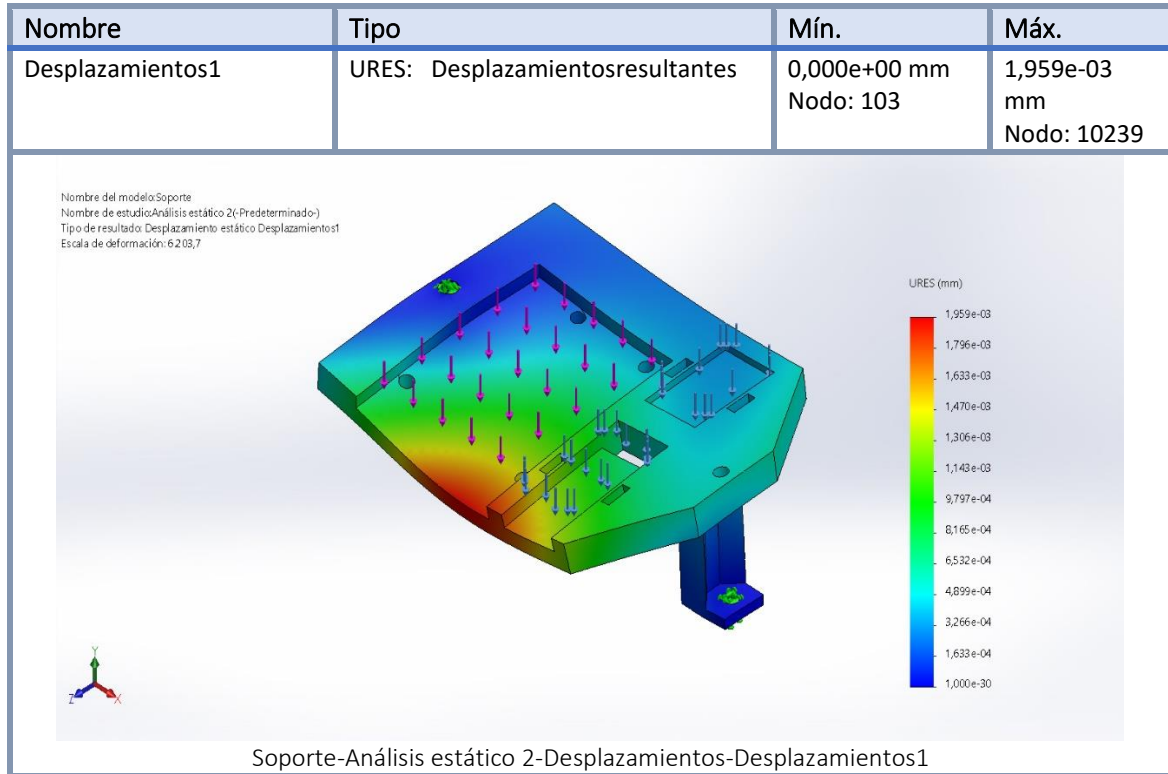


Tabla 28. Desplazamientos soporte superior

5.2. SOPORTE INFERIOR

- Análisis realizado: análisis estático
- Sujeciones: anclaje mediante pernos a la base
- Fuerzas aplicadas:
 - Arduino uno $\rightarrow 0,2646\text{N}$
 - Bomba y válvula de ventosa $\rightarrow 0,882\text{N}$
 - Pantalla LCD $\rightarrow 0,3038\text{N}$
 - PowerBank $\rightarrow 0,637\text{N}$

Nombre de carga	Cargar imagen	Detalles de carga
Fuerza-1		Entidades: 1 cara(s) Tipo: Aplicar fuerza normal Valor: 1,1466 N
Fuerza-2		Entidades: 1 cara(s) Tipo: Aplicar fuerza normal Valor: -0,637 N
Fuerza-3		Entidades: 2 cara(s) Referencia: Arista < 1 > Tipo: Aplicar fuerza Valores: ---; ---; 0,3038 N

Tabla 29. Fuerzas sobre el soporte inferior

- Mallado:

Número total de nodos	18298
Número total de elementos	10386
Cociente máximo de aspecto	12,692
% de elementos cuyo cociente de aspecto es < 3	96,5
% de elementos cuyo cociente de aspecto es > 10	0,0193
% de elementos distorsionados (Jacobiana)	0
Tiempo para completar la malla (hh:mm:ss):	00:00:01
Nombre de computadora:	

Tabla 30. Mallado soporte inferior

RESULTADOS DEL ESTUDIO

- Tensiones:

Tensión máxima = $1,546e+05 \text{ N/m}^2 < \text{Límite elástico} = 6,000e+07 \text{ N/m}^2$

Nombre	Tipo	Mín.	Máx.
Tensiones1	VON: Tensión de von Mises	8,709e+01 N/m ² Nodo: 1875	1,546e+05 N/m ² Nodo: 163

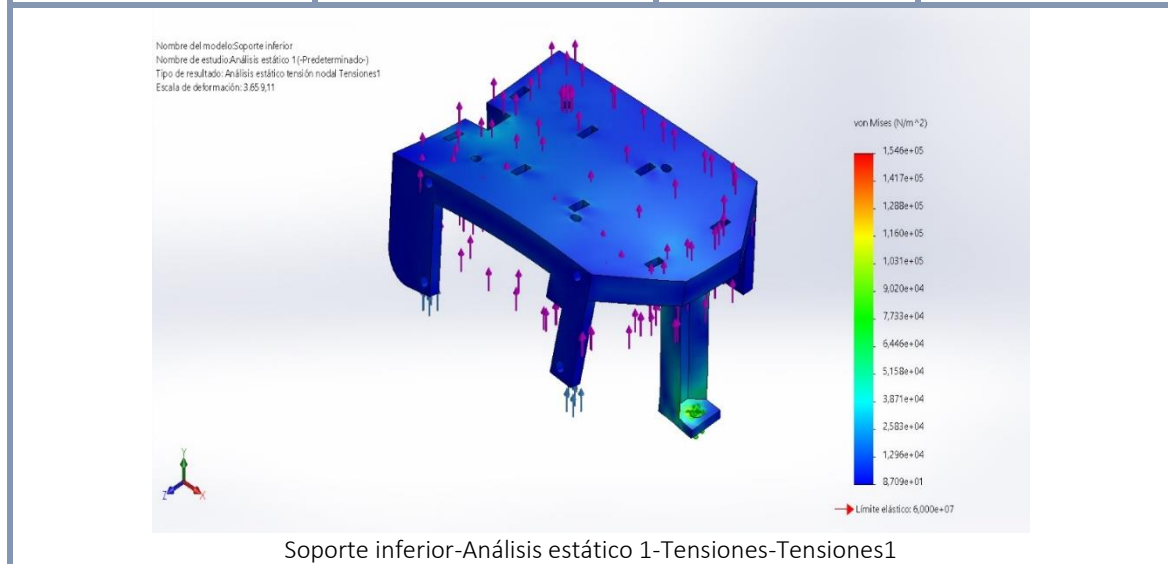


Tabla 31. Tensiones soporte inferior

- Desplazamientos:

Desplazamiento máximo = $3,425e-03 \text{ mm}$

Nombre	Tipo	Mín.	Máx.
Desplazamientos1	URES: Desplazamientos resultantes	0,000e+00 mm Nodo: 159	3,425e-03 mm Nodo: 12002

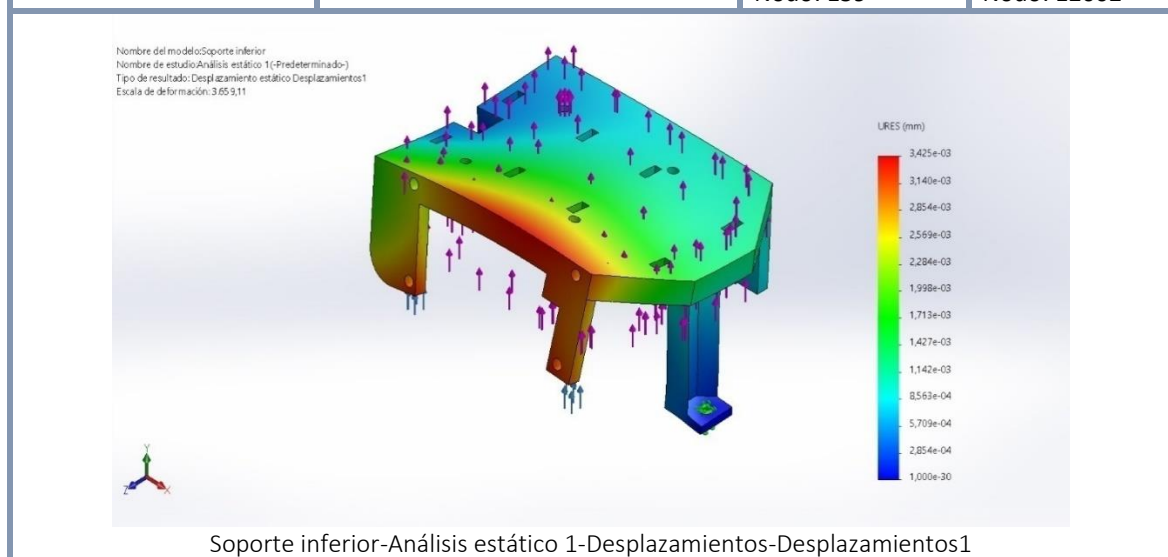


Tabla 32. Desplazamientos soporte inferior

5.3. CONJUNTO BASE:

El estudio realizado seguirá los siguientes puntos:

- Análisis realizado: análisis estático
- Sujeciones: anclaje mediante pernos a la muñeca del robot
- Sujeciones entre los componentes:
 - Se le aplicarán las sujeciones correspondientes a los pasadores que unen las distintas piezas que se encuentran en este modelo:

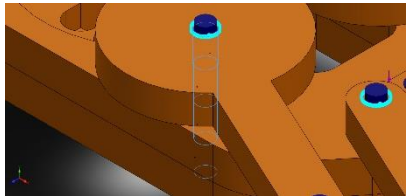
Referencia de modelo	Detalles del conector
 <p>Conectortipo Pasador-1</p>	Entidades: 4 cara(s) Tipo: Pasador Tipo de conexión: Con anillo de retención (sin traslación) Valor de rigidezrotacional: 0 Unidades: SI

Tabla 33. Sujeciones entre componentes del modelo de la base

- Se le aplica también la sujeción mediante el contacto de las dos caras enfrentadas de la base:

Contacto	Imagen del contacto	Propiedades del contacto
Contacto global		Tipo: Unión rígida Componentes: 1 componente(s) Opciones: Mallado compatible

Tabla 34. Contactos aplicados en el modelo de la base

- Fuerzas aplicadas [Tabla35]:

Peso correspondiente a los elementos soportados por la base superior	0,637 N
Soporte superior	0,5684N
Soporte inferior	0,7546N
Peso soportado por la base inferior	2,0874N
Servomotor	0,4214N
Pinzas	0,9604N
Ventosas	0,7252N
Soporte sensor de distancia y sensor	0,0686N

Tabla 35. Fuerzas aplicadas al conjunto de la base

- Sobre su cara superior se le aplicarán las fuerzas correspondientes al peso del soporte superior y los componentes que se encuentran unidos a él. Se le aplicará la fuerza sobre los dos puntos de apoyo del soporte con esta base [Figura Anexos 11]:

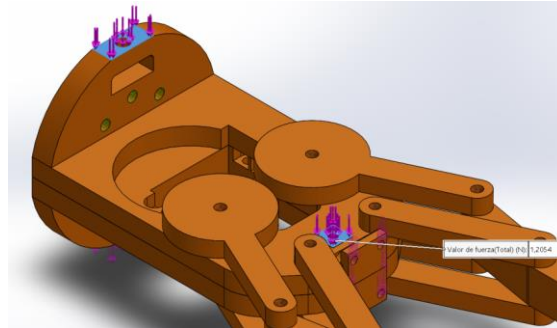


Figura Anexos 11. Fuerzas generadas por el soporte superior

- Sobre la cara inferior se le aplicarán 2 conjuntos de fuerzas distintas:
 - La carga correspondiente al soporte inferior y todos los componentes que van junto a él. Se aplica sobre los dos puntos de apoyo de este soporte [Figura Anexos 12].

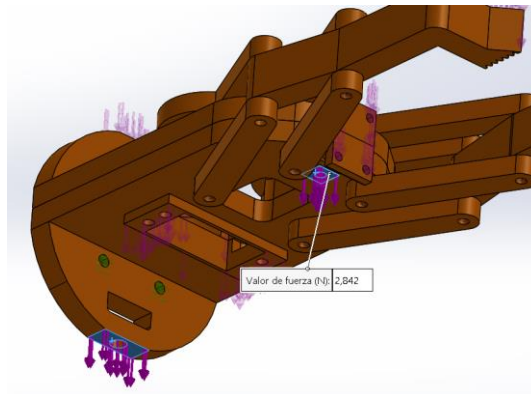


Figura Anexos 12. Cargas correspondientes al soporte inferior

- La fuerza generada por el peso del servomotor se aplica en la superficie en la que irá anclada el componente [Figura Anexos 13].

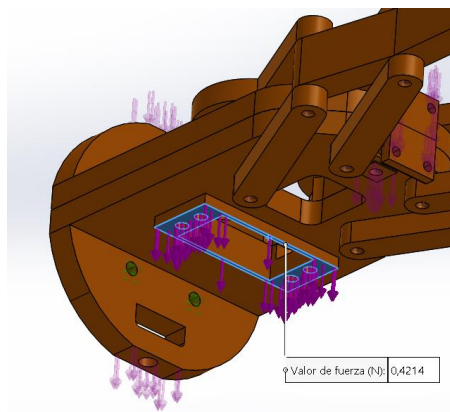


Figura Anexos 13. Cargas generadas por el servomotor

- Sobre la cara frontal de la herramienta se aplicará la fuerza correspondiente al peso de la herramienta ventosa, así como el del sensor de distancia [Figura Anexos 14].

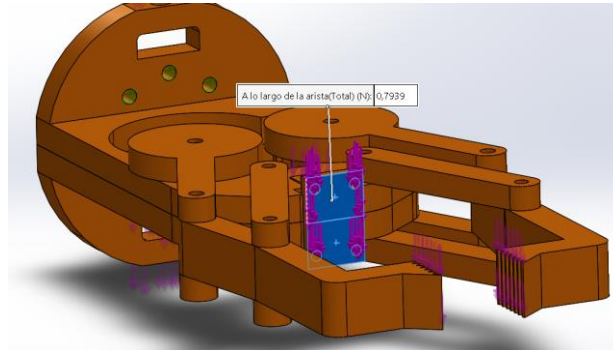


Figura Anexos 14. Cargas generadas por el sistema de ventosas y el sensor de distancia

- Finalmente, la carga del elemento que se manipulará mediante la pinza se coloca en el extremo frontal y en dirección hacia abajo [Figura Anexos 15]. De esta manera se simulará la situación más desfavorable a la que se puede ver sometida la herramienta.

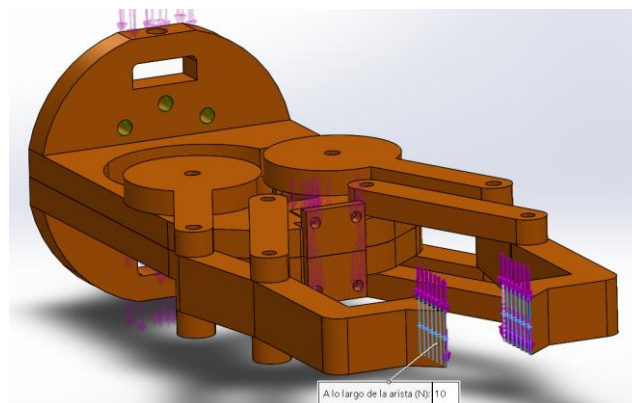


Figura Anexos 15. Carga generada por el elemento de manipulación

Se estudiará la herramienta con 3 valores distintos de esta fuerza para determinar cuál ha de ser la carga máxima de utilización de esta:

- Estudio 1: 1kg; aproximación a 10N
- Estudio 2: 2kg; aproximación a 20N
- Estudio 3: tras realizar el estudio con un gran rango de pesos, el tercer estudio que se mostrará dentro de este proyecto corresponde al límite de peso que se ha considerado como máximo para la utilización de la herramienta.

RESULTADOS DEL ESTUDIO PARA 1 KG

- Tensión máxima = $2,739e+07 \text{ N/m}^2$ < Límite elástico = $6,000e+07 \text{ N/m}^2$

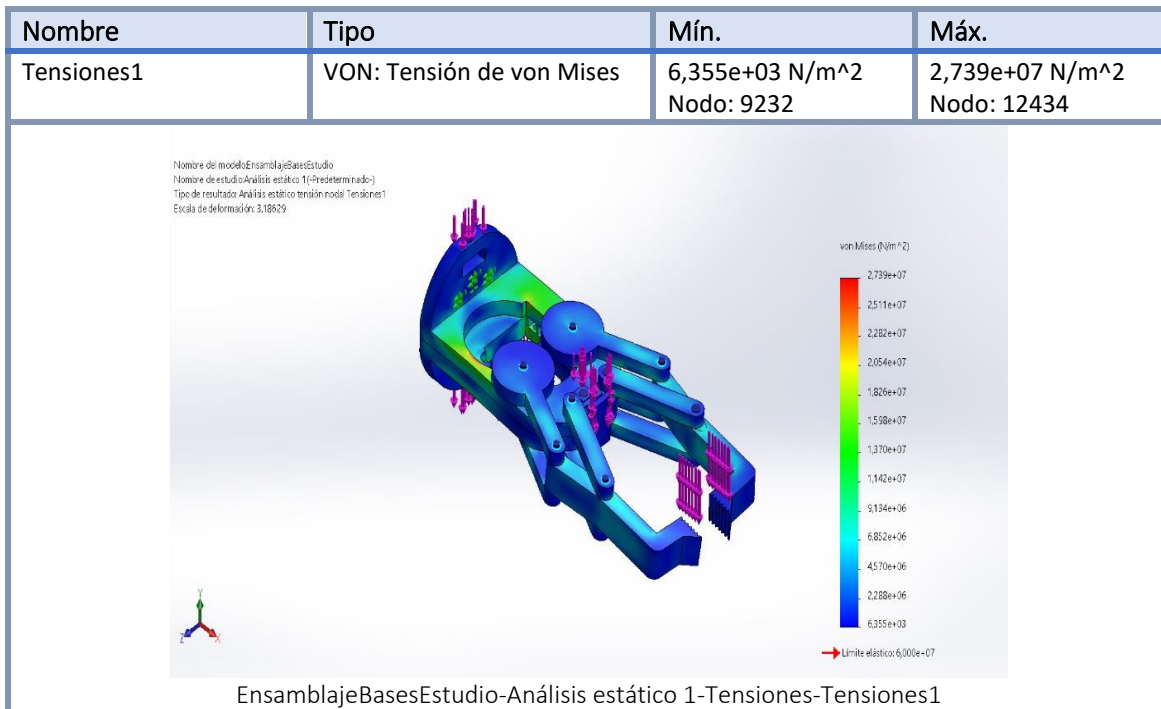


Tabla 36. Tensiones en el conjunto base para manipulación de 1Kg

- Desplazamiento máximo = 7,82 mm.

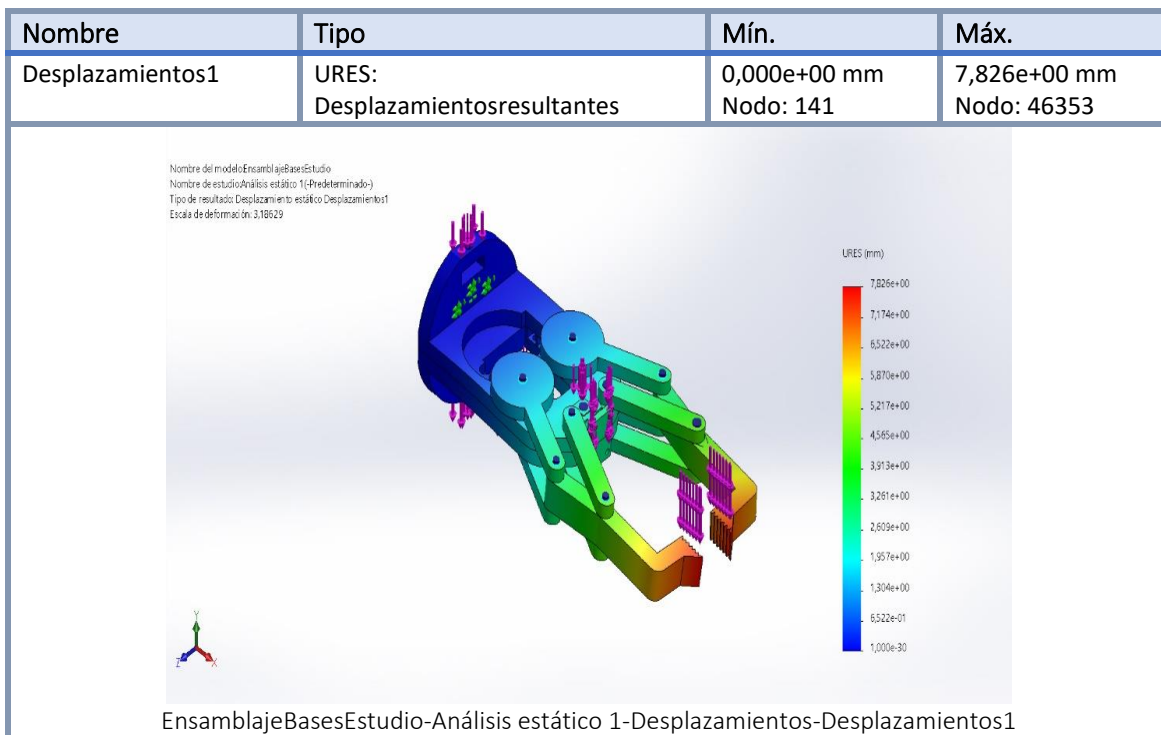


Tabla 37. Desplazamientos en el conjunto base para manipulación de 1Kg

RESULTADOS DEL ESTUDIO PARA 2 KG

- Tensión máxima = $5,436e+07 \text{ N/m}^2$ < Límite elástico = $6,000e+07 \text{ N/m}^2$

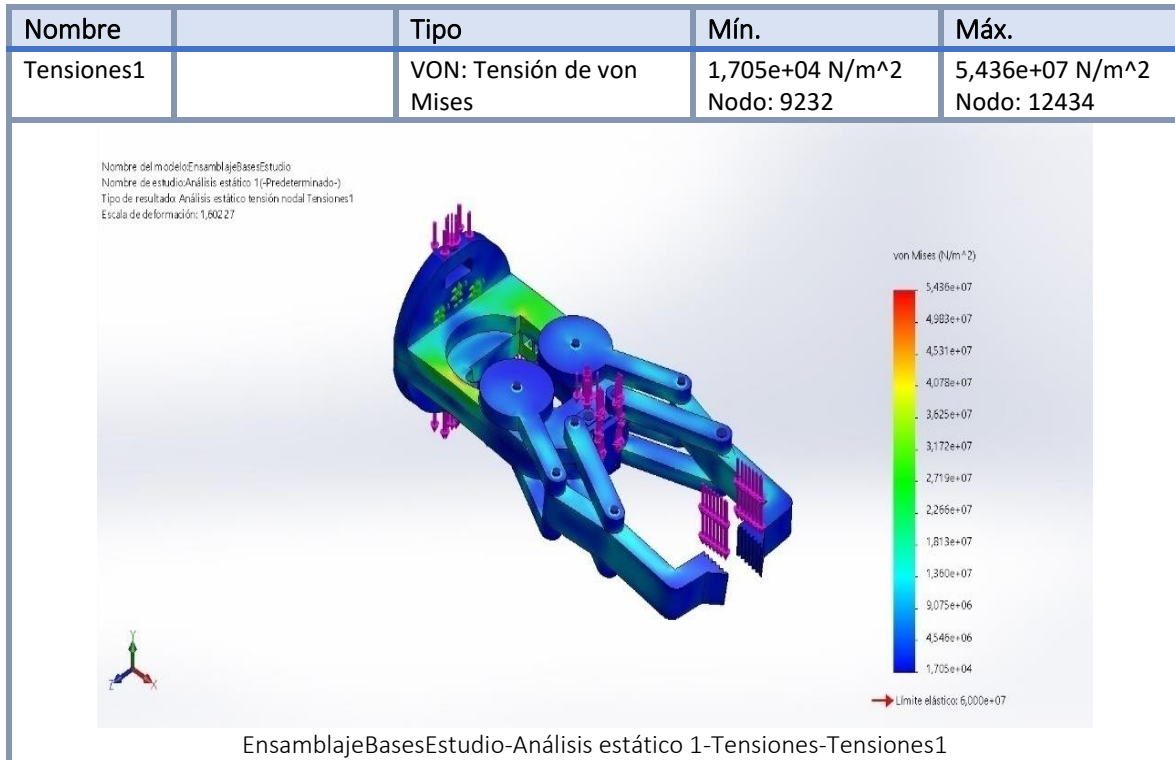


Tabla 38. Tensiones en el conjunto base para manipulación de 2Kg

- Desplazamiento máximo = 15,5 mm.

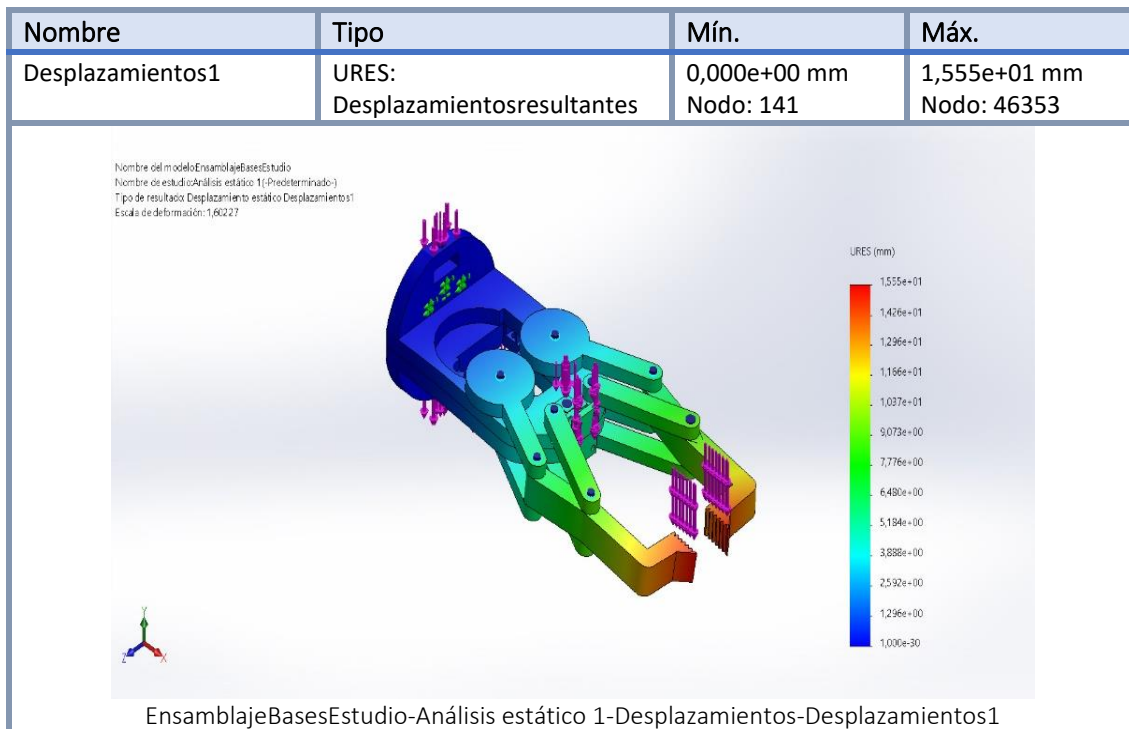


Tabla 39. Desplazamientos en el conjunto base para manipulación de 2Kg

RESULTADOS DEL ESTUDIO PARA 500 G: PESO MÁXIMO DE USO

- Tensión máxima = $1,386e+07 \text{ N/m}^2$ < Límite elástico = $6,000e+07 \text{ N/m}^2$

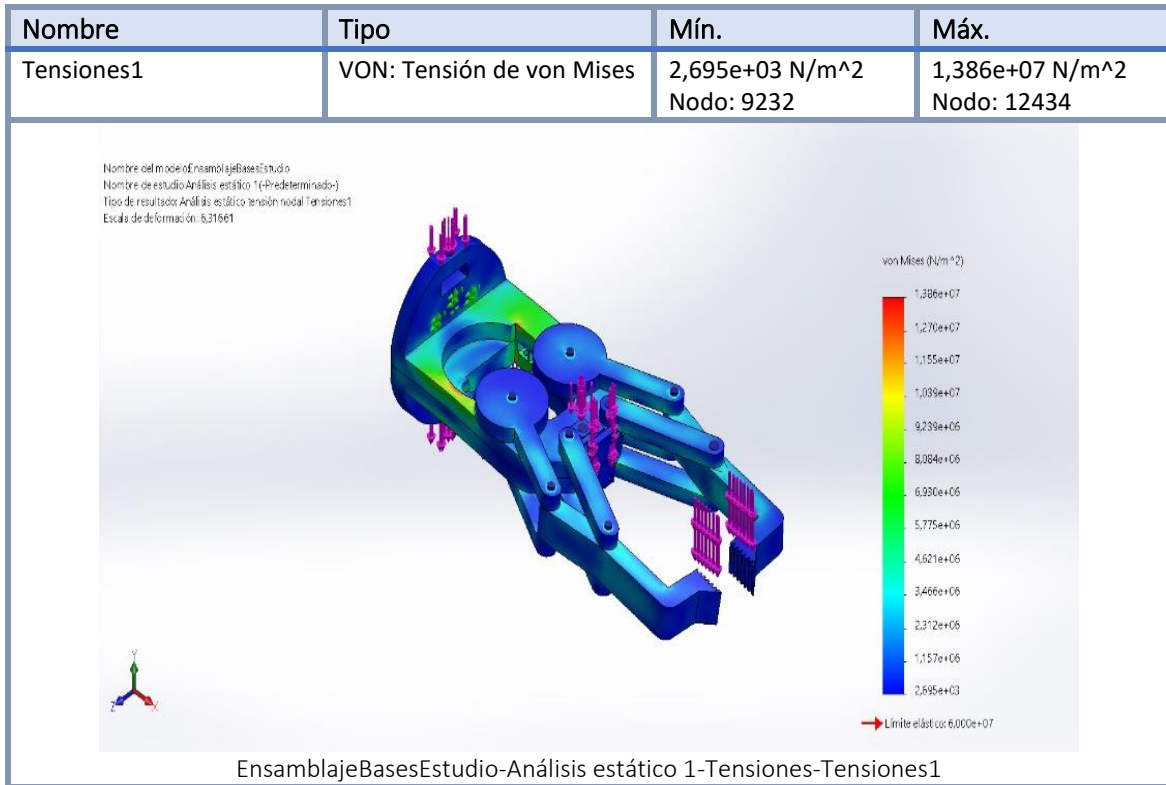


Tabla 40. Tensiones en el conjunto base para manipulación de 0.5Kg

- Desplazamiento máximo = 3,95 mm.

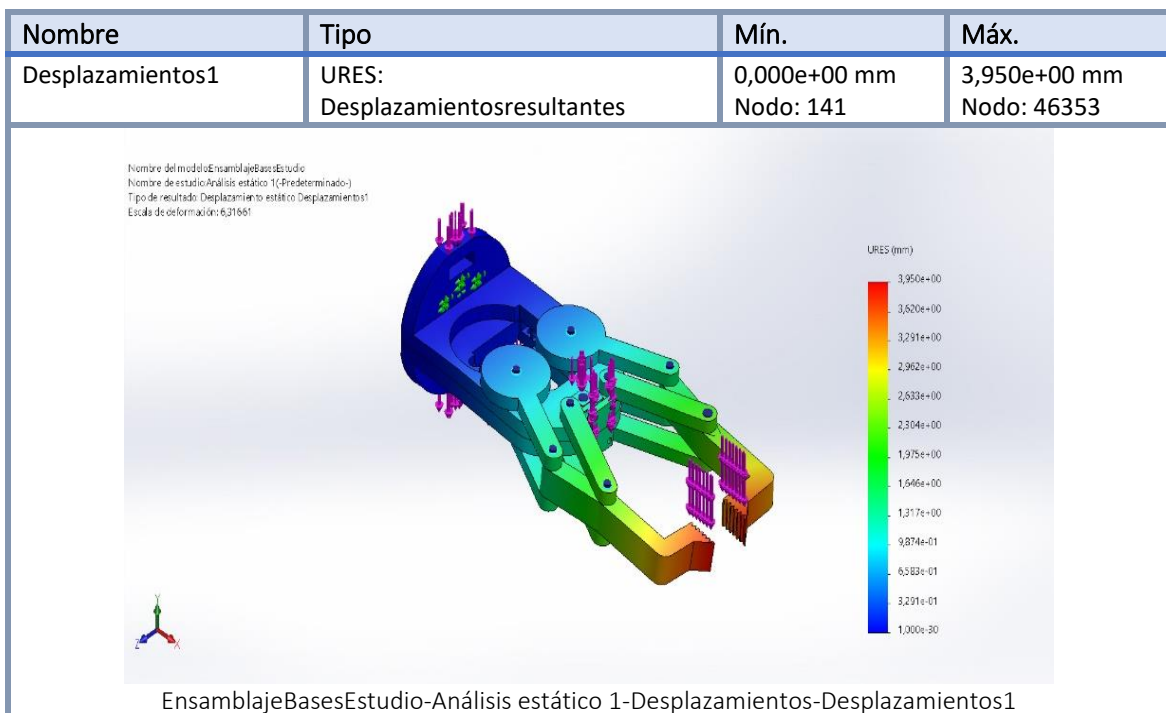


Tabla 41. Desplazamientos en el conjunto base para manipulación de 0.5Kg



Diseño y fabricación de una herramienta multifunción para un robot industrial.

ANEXO 2: CÓDIGOS

FASE 1: ARDUINO NANO 1 + MODULO COMUNICACIÓN (CONJUNTO EMISOR)

```
#include<nRF24L01.h>
#include<RF24.h>
#include<RF24_config.h>
#include<SPI.h>

constintpinCE= 9;
constintpinCSN= 10;
RF24 radio(pinCE,pinCSN);

constint inputPinza180 =2; //Pin que activa abrir pinza
constint inputPinza44 =3; //Pin que activa cerrar pinza a 44mm
constint inputPinza88 =4; //Pin que activa cerrar pinza a 88mm
constintinputVentosaSuc=5; //Pin que activa la succion de la ventosa
constintinputVentosaAguanta=6; //Pin que activa que la ventosa
aguante la succion
constintinputVentosaSuelta=7; //Pin que activa soltar presion de la
ventosa

byte v2=0; //valor del pin 2 (correspondiente a abrir pinza)
byte v3=0; //valor del pin 3 (correspondiente a cerrar pinza 44)
byte v4=0; //valor del pin 4 (correspondiente a cerrar pinza 88)
byte v5=0; //valor del pin 5 (correspondiente a succion de la ventosa)
byte v6=0; //valor del pin 6 (correspondiente a aguantar succion)
byte v7=0; //valor del pin 7 (correspondiente a fin succion)

// Single radio pipe addressforthe 2 nodestocommunicate.
constuint64_t pipe = 0xE8E8F0F0E1LL;

intdata[1];

voidsetup(void)
{
    radio.begin();
    radio.openWritingPipe(pipe);
    pinMode(inputPinza180, INPUT);
    pinMode(inputPinza44, INPUT);
    pinMode(inputPinza88, INPUT);
    pinMode(inputVentosaSuc, INPUT);
    pinMode(inputVentosaAguanta, INPUT);
    pinMode(inputVentosaSuelta, INPUT);
}

voidloop(void)
{
    data[0]= 0;

    v2 =digitalRead(inputPinza180); //Lee la entrada del PIN 2 y lo
guarda en v2
    v3 =digitalRead(inputPinza44); //Lee la entrada del PIN 3 y lo
guarda en v3
    v4 =digitalRead(inputPinza88); //Lee la entrada del PIN 4 y lo
guarda en v4
```

```
v5 =digitalRead(inputVentosaSuc); //Lee la entrada del PIN 5 y lo
guarda en v5
v6 =digitalRead(inputVentosaAguanta); //Lee la entrada del PIN 6 y
lo guarda en v6
v7 =digitalRead(inputVentosaSuelta); //Lee la entrada del PIN 7 y
lo guarda en v7

if (v2 ==HIGH&& v3==LOW&& v4==LOW) { //Cuando la salida 2 esta
activa (y la 3 y 4 no) quiero abrir la pinza a 180° y envio un 1
  data[0]=1; //valor que se envia
  radio.write(data,sizeof data); //envio mediante radio
  delay(1000); //tiempo de espera para enviar un valor cada 1000
milisegundos
}

elseif(v3 ==HIGH&& v2==LOW&& v4==LOW){ //Cuando la salida 3 (y la 2
y 4 no) esta activa quiero cerrar la pinza a 44mm y envio un 2
  data[0]=2;
  radio.write(data,sizeof data);
  delay(1000);
}

elseif(v4 ==HIGH&& v2==LOW&& v3==LOW){ //Cuando la salida 4 esta
activa (y la 3 y 2 no)quiero cerrar la pinza a 88mm y envio un 3
  data[0]=3;
  radio.write(data,sizeof data);
  delay(1000);
}

elseif(v5 ==HIGH){ //Cuando la salida 5 esta activa Succion con
la ventosa, envio un 4
  data[0]=4;
  radio.write(data,sizeof data);
  delay(1000);
}

elseif(v6 ==HIGH){ // Cuando la salida 6 esta activa Aguanto el
vacio, envio un 5
  data[0]=5;
  radio.write(data,sizeof data);
  delay(1000);
}

elseif(v7 ==HIGH){ // Cuando la salida 7 esta activa Aguanto el
vacio, envio un 6
  data[0]=6;
  radio.write(data,sizeof data);
  delay(1000);
}

else{ //Si no, envio 0 que no repercuten en mi codigo
  data[0]=0;
  radio.write(data,sizeof data);
  delay(1000);
}
}
```

FASE 2: ARDUINO NANO (2) + MODULO COMUNICACIÓN

// en este programa el arduino lee lo que le envia el otro ard nano mediante radio y lo manda por puerto serie al arduino uno.

```
#include<SoftwareSerial.h>
#include<nRF24L01.h>
#include<RF24.h>
#include<RF24_config.h>
#include<SPI.h>

constintpinCE= 9;
constintpinCSN= 10;
RF24 radio(pinCE,pinCSN);

// Single radio pipe addressforthe 2 nodestocommunicate.
constuint64_t pipe = 0xE8E8F0F0E1LL;

intdata[1];

voidsetup(){
  Serial.begin(9600);
  radio.begin();
  radio.openReadingPipe(1,pipe);
  radio.startListening();
}

voidloop()
{
  if (radio.available())
  {
    radio.read(data,sizeof data);
    Serial.print(data[0]);
  }
  delay(1000);
}
```

FASE 3: ARDUINO UNO (1) + SENSOR SHIELD

```
#include<Servo.h>

// Declaramos la variable para controlar el servo
Servo servo1;
Servo servo2;
Servo servo3;
int dato;

voidsetup() {

//Inicio los pines donde van los distitntos servos
servo1.attach(6); //-> Servo pinza
servo2.attach(5); // -> Bomba
servo3.attach(4); // ->Valvula

//Inicio la comunicacion con el puerto serie
Serial.begin(9600);

//Pongo todo en su posicion 0
servo1.write(180); //Pinza abierta
servo2.write(0); //Ventosa sin presion
servo3.write(0); //Ventosa sin presion
}

voidloop() {
  dato=Serial.read();
  switch(dato){ //Comprobamos el dato

    case'1': //Si recibimos un '1' abro total la pinza...
      servo1.write(180);
      break;

    case'2': //si recibimos una '2' cierro la pinza a 44mm
para coger pieza grande
      servo1.write(42,1);
      break;

    case'3': //si recibimos una '3' cierro la pinza a 88mm
para coger pieza pequeña
      servo1.write(99,58);
      break;

    case'4': //si recibimos una '4' Ventosa hace vacio
servo2.write(180);
servo3.write(0);
//EN TODAS ESTAS LA PINZA DEBE ESTAR ABIERTA DEL TODO,
no molestar a la ventosa
servo1.write(180);
      break;

    case'5': //si recibimos una '5' la ventosa Aguantar el
vacio
      servo2.write(0);
      servo3.write(0);
      servo1.write(180);
      break;
  }
}
```



Diseño y fabricación de una herramienta multifunción para un robot industrial.

```
case '6': //si recibimos una '6' la ventosa suelta el vacio
    servo2.write(0);
    servo3.write(180);
    servo1.write(180);
    break;
}
}
```

FASE 4: ARDUINO UNO (2)

```
// SVL53L0X Sensor de distancia

#include "Adafruit_VL53L0X.h"
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// Configuración de los caracteres de pantalla
LiquidCrystal_I2C lcd(0x3F, 16, 2);

Adafruit_VL53L0X lox = Adafruit_VL53L0X();

void setup() { //inicio puertos de la pantalla y puerto serie
  lcd.init();
  lcd.backlight();
  Serial.begin(9600);

  // Esperar puerto serie
  while (! Serial) {
    delay(1);
  }

  Serial.println("Test Adafruit VL53L0X");
  if (!lox.begin()) {
    Serial.println(F("Fallo boot VL53L0X"));
    while(1);
  }
  // power
  Serial.println(F("VL53L0X API Simple Ranging example\n\n"));
}

void loop() {
  VL53L0X_RangingMeasurementData_t measure;

  Serial.print("Leyendo medidas... ");
  lox.rangingTest(&measure, false); // 'verdadero' para obtener la
  impresión de los datos de depuración

  if (measure.RangeStatus != 4) {
    Serial.print("Distancia (mm): ");
    Serial.println(measure.RangeMilliMeter);
    lcd.clear(); //
    lcd.setCursor(0,0);
    lcd.print(" DISTANCIA en mm ");
    lcd.setCursor(0,1);
    lcd.print(measure.RangeMilliMeter);
  } else {
    Serial.println(" Fuera rango ");
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" DISTANCIA en mm ");
    lcd.setCursor(0,1);
    lcd.print(" Fuera rango ");
  }

  delay(100);
}
```


PROGRAMACIÓN IRC5C RAPID

MODULE Module1

```
CONST robtarget Target_10:=[[0,-
100,100],[0,0,1,0],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget Target_20:=[[ -80,148,-30],[0.5,0.5,0.5,-
0.5],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget Target_30:=[[0,300,-110],[0.5,0.5,0.5,-
0.5],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget Target_40:=[[80,148,-30],[0.5,0.5,0.5,-
0.5],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget Target_50:=[[444,105,-110],[0.5,0.5,0.5,-
0.5],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget Target_60:=[[ -444,105,-110],[0.5,0.5,0.5,-
0.5],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget
Home:=[[662.176223927,0,589.5],[0.5,0,0.866025404,0],[0,0,0,0],[9E+09,9E+09,9E+0
9,9E+09,9E+09,9E+09]];
CONST robtarget Target_80:=[[400,400,-110],[0.5,0,0.5,-
0.707106781],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget Target_70:=[[ -400,400,-110],[0.707106781,0.5,0,-
0.5],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget
Inicio_10:=[[662.176223927,0,589.5],[0.5,0,0.866025404,0],[0,0,0,0],[9E+09,9E+09,9
E+09,9E+09,9E+09,9E+09]];
CONST robtarget Target_90:=[[80,215,-190],[0.5,0.5,0.5,-
0.5],[0,0,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
CONST robtarget Target_100:=[[ -80,218,-190],[0.5,0.5,0.5,-0.5],[ -1,0,-
1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
```

```
!*****
! Módulo: Module1
! Descripción:
! Programa para la prueba de trayectoria del robot usando pinza
! Autor: Borja Hierro
! Versión: 1.0
!*****
```

PROC INICIO()

```
MoveL Inicio_10,v100,z100,TCPPinza_Ventosa\WObj:=wobj0;
ENDPROC
```

PROC Ir_a_coger_disco()

```
MoveJ Inicio_10,v100,z100,TCPPinza_Ventosa\WObj:=wobj0;
MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;
```

```
MoveJ Target_40,v100,fine,TCPPinza_Ventosa\WObj:=wobj_discos;
ENDPROC
PROC Ir_a_dejar_disco()
MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;
MoveJ Target_20,v100,fine,TCPPinza_Ventosa\WObj:=wobj_discos;
ENDPROC
PROC Ir_a_coger_pieza_grande()
MoveJ Target_80,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;
MoveJ Target_50,v100,fine,TCPPinza_Ventosa\WObj:=wobj_discos;
ENDPROC

PROC Ir_a_dejar_pieza_grande()
MoveJ Target_70,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;
MoveJ Target_60,v100,fine,TCPPinza_Ventosa\WObj:=wobj_discos;
ENDPROC

PROC Ir_a_coger_pequeño()
MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;
MoveJ Target_90,v100,fine,TCPPinza_Ventosa\WObj:=wobj_discos;
ENDPROC

PROC Ir_a_dejar_pequeño()
MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;
MoveJ Target_100,v100,fine,TCPPinza_Ventosa\WObj:=wobj_discos;
ENDPROC

!*****
! Procedimiento Main
!*****

PROC main()
!WaitDI Armar,1;
INICIO;
!Reseteo todas las salidas
Reset DO11_2;
Reset DO11_3;
Reset DO11_4;
Reset DO11_5;
Reset DO11_6;
Reset DO11_7;

!Set y reset de DO11_2 para abrir la pinza
set DO11_2;
WaitTime 1;
Reset DO11_2;

!Ciclo coger pieza grande (pinza cierra a 88mm)

Ir_a_coger_pieza_grande;
WaitTime 2;
```

Set DO11_4; !Activo la salida de cerrar pinza a 88mm
WaitTime 3;
MoveJ Target_80,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;
INICIO;

!Pieza grande cogida, voy a dejarla
Ir_a_dejar_pieza_grande;
WaitTime 2;
Reset DO11_4;!Pongo a 0 la salida de cerrar pinza a 88mm
Set DO11_2; !Activo la salida que abre la pinza
WaitTime 3;
MoveJ Target_70,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;
INICIO;
Reset DO11_2; !reseteo la salida de pinza abierta

!Ciclo coger discos

Ir_a_coger_disco;
WaitTime 1;
Set DO11_5; !Activo la salida de succion con la ventosa
WaitTime 4; !4 segundos de succion
reSet DO11_5; !Desactivo la succion
Set DO11_6; !Activo la variable que mantiene el vacio y la deajo
MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;

!Pieza cogida con vacio, voy a dejarla
Ir_a_dejar_disco;
WaitTime 2;
Reset DO11_6;!La ventosa suelta el vacio
Set DO11_7;!La válvula suelta la pieza
WaitTime 3;
Reset DO11_7;
MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;
INICIO;

!Ciclo coger pieza pequeña (pinza cierra a 44mm)

Ir_a_coger_pequeño;
WaitTime 1;
Set DO11_3; !Activo la salida de cerrar pinza a 44mm
WaitTime 4;
MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;

!Pieza pequeña cogida, voy a dejarla
Ir_a_dejar_pequeño;
WaitTime 2;
Reset DO11_3;!Pongo a 0 la salida de cerrar pinza a 88mm
Set DO11_2; !Activo la salida que abre la pinza
WaitTime 3;
MoveJ Target_30,v100,z100,TCPPinza_Ventosa\WObj:=wobj_discos;



Diseño y fabricación de una herramienta multifunción para un robot industrial.

INICIO;
Reset DO11_2; !reseteo la salida de pinza abierta

!Discos;

ENDPROC

ENDMODULE



DOCUMENTO 2: FASES TFM Y CRONOGRAMA



Diseño y fabricación de una herramienta multifunción para un robot industrial.

1. FASES Y CRONOGRAMA

1.1. FASES DEL TFM

A la hora de la realización de este TFM se han distinguido las distintas fases que se enumeran a continuación:

- **Tarea 1: Obtención de información de los sistemas involucrados dentro de la herramienta que se pretende llevar a cabo.** Se realiza una fase de información de los distintos subsistemas que posteriormente van a tener que ser incluidos en la herramienta o utilizados para su fabricación y puesta en funcionamiento: brazo robótico ABB, fabricación mediante impresión 3D, servomotores, programación mediante Arduino, tipos y formas de herramientas existentes para el uso mediante robots, etc.
- **Tarea 2: estudio de los requisitos de los requisitos de diseño de la herramienta.**
- **Tarea 3: Diseño y validación de los elementos mecánicos.** Se realiza un minucioso proceso de diseño de cada uno de los elementos que compondrán la herramienta. Se pone especial detalle en el cumplimiento de los requisitos de diseño y en la viabilidad de su posterior funcionamiento conjunto.
- **Tarea 4: fabricación de los elementos diseñados.** Se estudia el método de fabricación mediante impresión 3D y se obtienen las piezas diseñadas en formato físico. Se comprueba su correcto resultado y su funcionamiento conjunto.
- **Tarea 5: Obtención de un sistema de control para el conjunto formado por el robot y la herramienta.** Mediante la programación de los distintos elementos involucrados, se alcanza un sistema de control que cumpla con todos los requisitos iniciales de diseño.
- **Tarea 6: Simulación del sistema a través de Software.** Mediante la herramienta RobotStudio se verifica el correcto funcionamiento del sistema.
- **Tarea 7: Puesta en marcha del sistema real.** Dentro de las instalaciones de la universidad de Almería se realizan las pruebas correspondientes para verificar el correcto funcionamiento del sistema final.
- **Tarea 8: realización de planos y presupuestos.** Se obtienen todos los planos de cada uno de los elementos de la herramienta. Se estima el coste total de la producción de este sistema.

- **Tarea 9: Redacción de la memoria.** Se crea el presente trabajo, donde se agrupan todas las experiencias en la realización de este.

En el apartado siguiente se detallan cada una de las tareas por medio de un cronograma.

1.2. CRONOGRAMA

Dentro del cronograma que se muestra a continuación se han dividido los meses en quincenas y en él se distribuyen las horas empleadas a cada una de las tareas expuestas en el punto anterior. Se ha seguido un orden cronológico desde el mes de diciembre en el que se comenzó el proyecto hasta la fecha final de entrega.

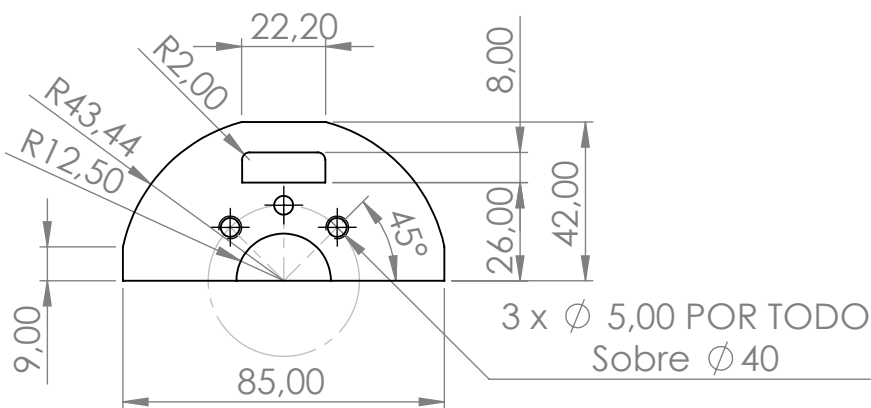
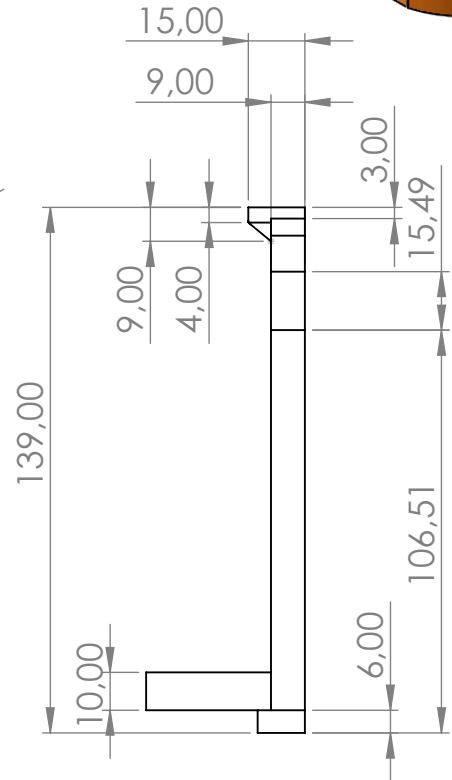
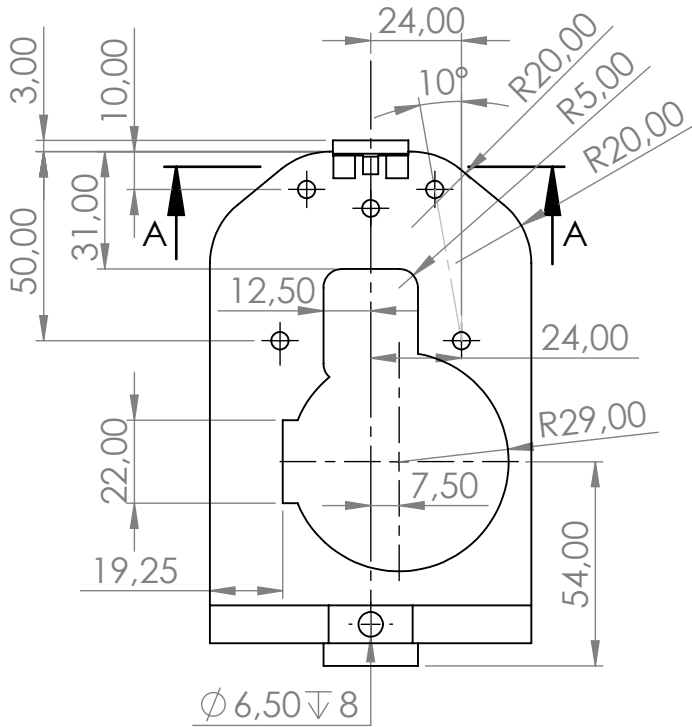
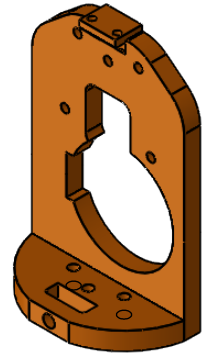
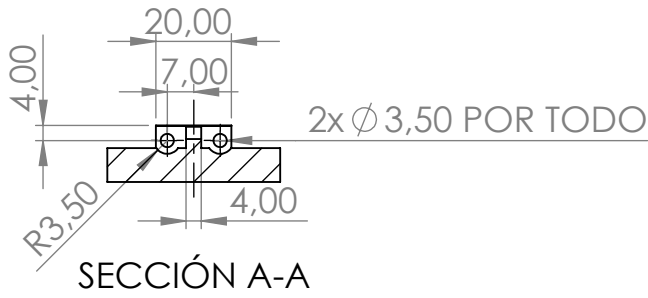
	HORAS	DICIEMBRE	ENERO	FEBRERO	MARZO	ABRIL	MAYO
TAREA 1	35						
TAREA 2	15						
TAREA 3	55						
TAREA 4	45						
TAREA 5	45						
TAREA 6	35						
TAREA 7	30						
TAREA 8	25						
TAREA 9	100						

Dentro del cronograma se puede observar que la mayoría de las tareas se han podido realizar de forma simultánea unas con otras, sin embargo, han existido tareas que no se han podido comenzar hasta que las anteriores se encontrasen ya finalizadas, como es el caso de la tarea 7 de puesta en funcionamiento del sistema real.

La suma total de las horas empleadas en la realización de este TFM es de 385 horas repartidas a lo largo de 6 meses.

DOCUMENTO3: PLANOS.

Taladros $\phi 4,60$ salvo cota especificada



SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM

TFM: Diseño y fabricación de una herramienta multifunción para un robot industrial

Master Ingeniería industrial
Universidad de Almería

AUTOR: BORJA HIERRO GARRIDO



PESO: 54g

DESCRIPCIÓN: Mitad superior de la base de la herramienta

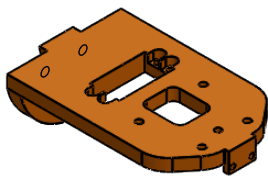
MATERIAL:
PLA

N.º DE DIBUJO
BASE superior

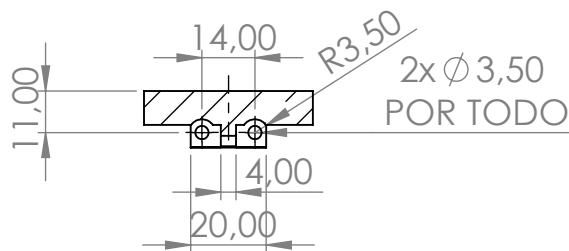
A4

ESCALA:1:2

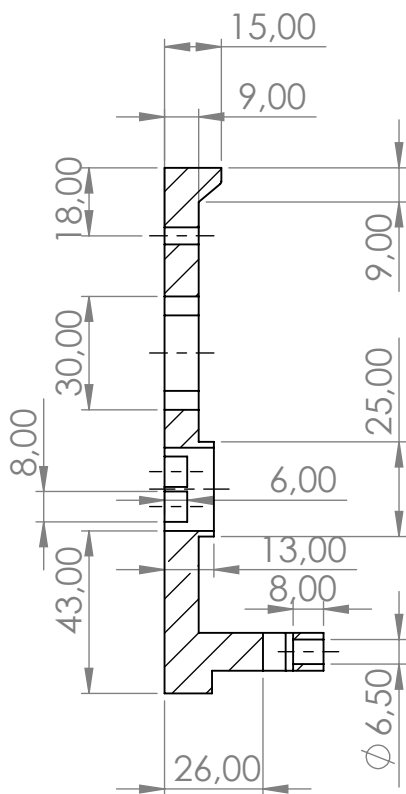
HOJA 1 DE 1



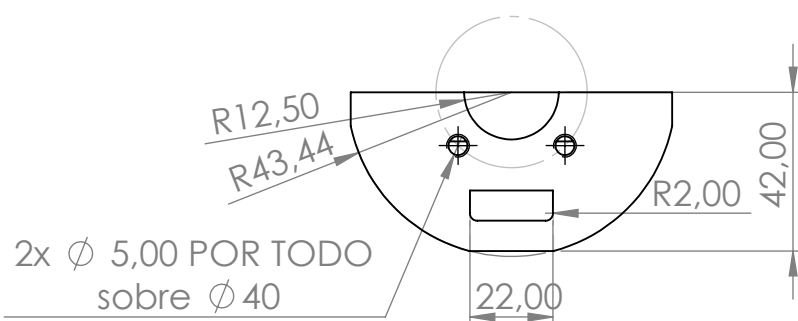
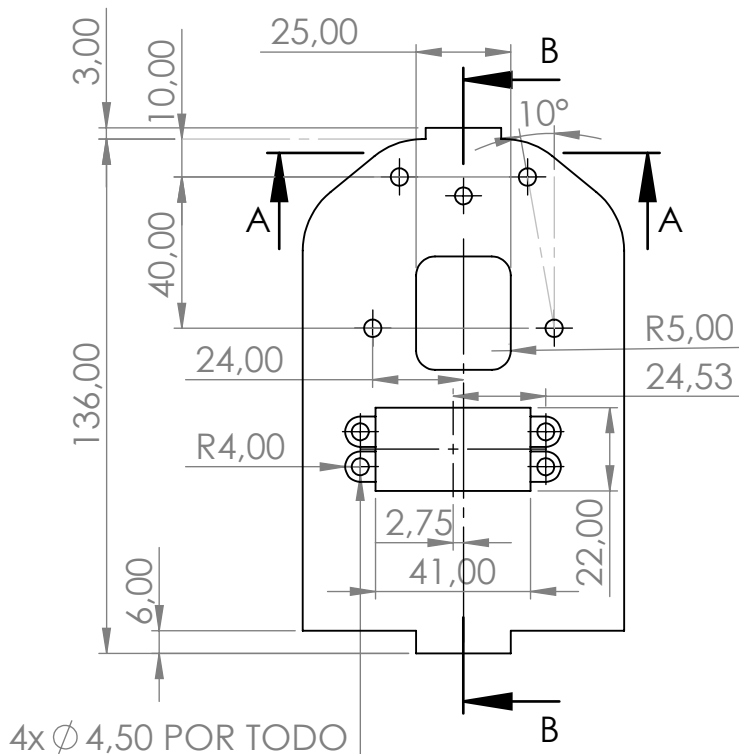
Taladros $\varnothing 4,60$ salvo cota especificada



SECCIÓN A-A



SECCIÓN B-B



SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM

TFM: Diseño y fabricación de una herramienta multifunción para un robot industrial

Master Ingeniería industrial
Universidad de Almería

AUTOR: BORJA HIERRO GARRIDO



PESO: 64g

DESCRIPCIÓN: Mitad inferior de la base de la herramienta

MATERIAL:
PLA

N.º DE DIBUJO

BASE inferior

A4

ESCALA:1:2

HOJA 1 DE 1

4 3 2 1

F

F

E

E

D

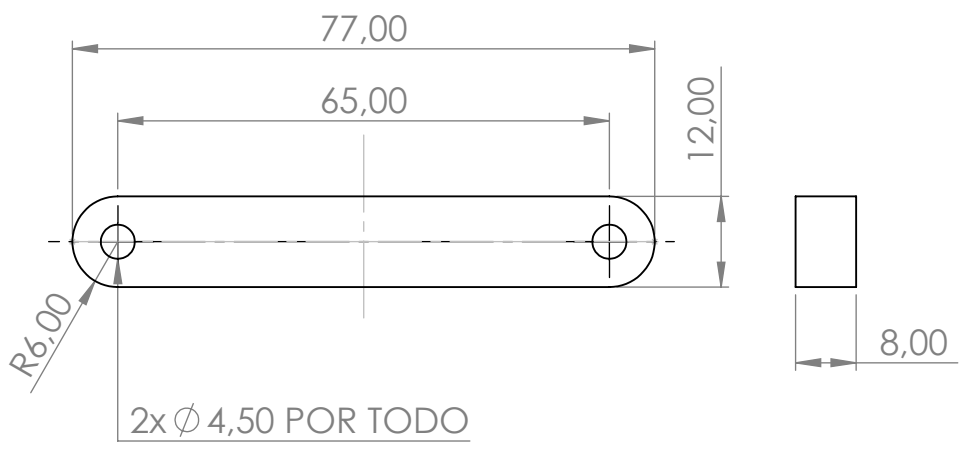
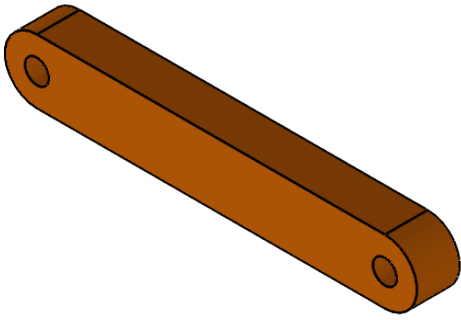
D

C

C

B

B



SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM

TFM: Diseño y fabricación de una herramienta multifunción para un robot industrial

Master Ingeniería industrial
Universidad de Almería

AUTOR: BORJA HIERRO GARRIDO



PESO: 6g

DESCRIPCIÓN: Elemento simple del mecanismo de 4 barras que forma el sistema pinza

MATERIAL:
PLA

N.º DE DIBUJO
Eslabon simple

A4

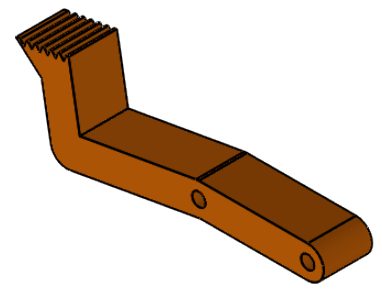
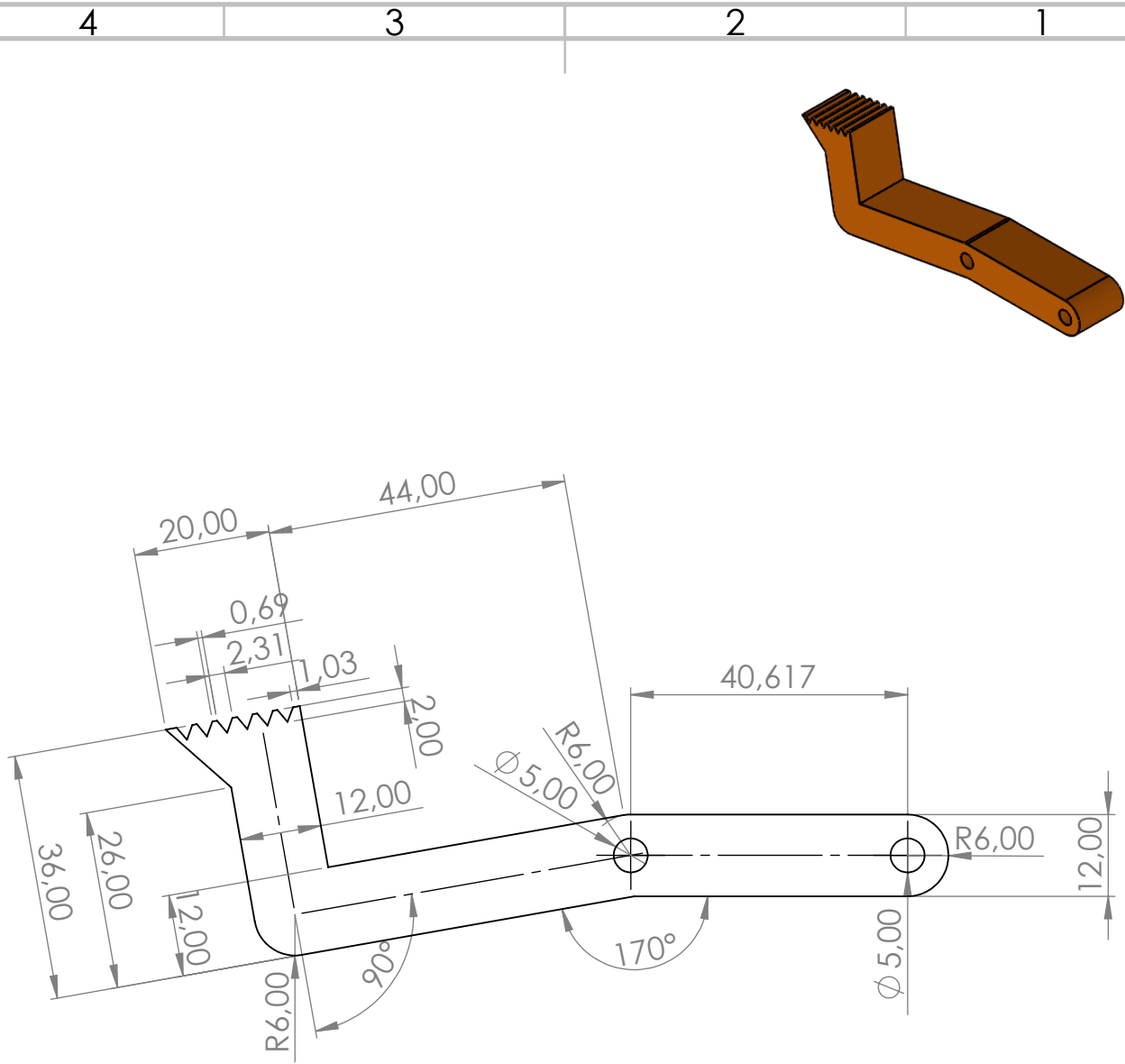
ESCALA:1:1

HOJA 1 DE 1

4 3 2 1

A

A



SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM

TFM: Diseño y fabricación de una herramienta multifunción para un robot industrial

Master Ingeniería industrial
Universidad de Almería

AUTOR: BORJA HIERRO GARRIDO



PESO: 17g

DESCRIPCIÓN: Elemento final de agarre del mecanismo de 4 barras que forma el sistema pinza

MATERIAL:
PLA

N.º DE DIBUJO
Eslabón pinza

A4

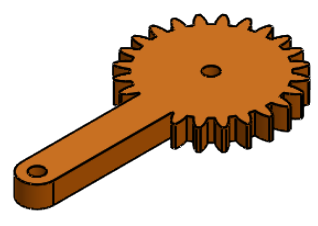
ESCALA:1:1

HOJA 1 DE 1

4 3 2 1

F

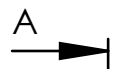
F



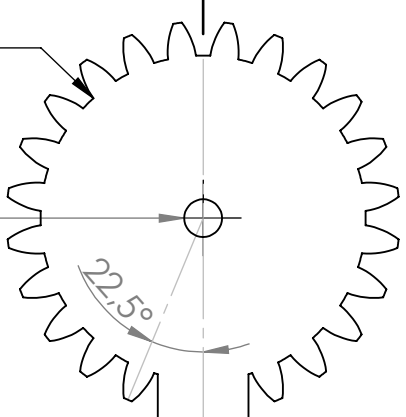
E

E

Características engranaje:
 Diámetro de paso=48mm
 Número de dientes =24
 Módulo=2



Ø 5,00 POR TODO



Ø 5,00 POR TODO

12,00

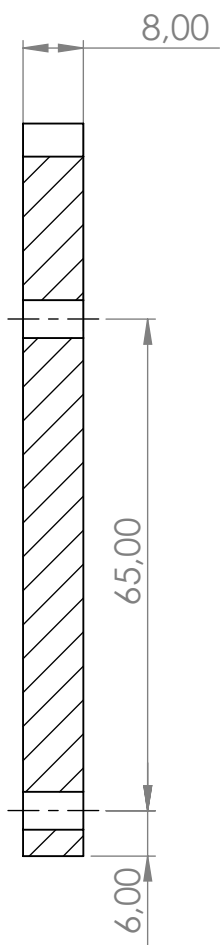
R6,00

C

C

B

B



SECCIÓN A-A
 ESCALA 1 : 1

SI NO SE INDICA LO CONTRARIO:
 LAS COTAS SE EXPRESAN EN MM

TFM: Diseño y fabricación de una herramienta multifunción para un robot industrial

Master Ingeniería industrial
 Universidad de Almería

AUTOR: BORJA HIERRO GARRIDO

A

A



PESO: 14g

DESCRIPCIÓN: Eslabon del sistema 4 barras con engranaje para acople al servomotor

MATERIAL:
PLA

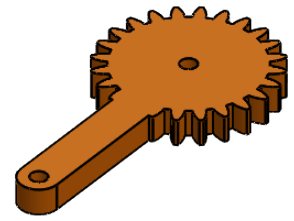
N.º DE DIBUJO
Eslabón tipo pinza

A4

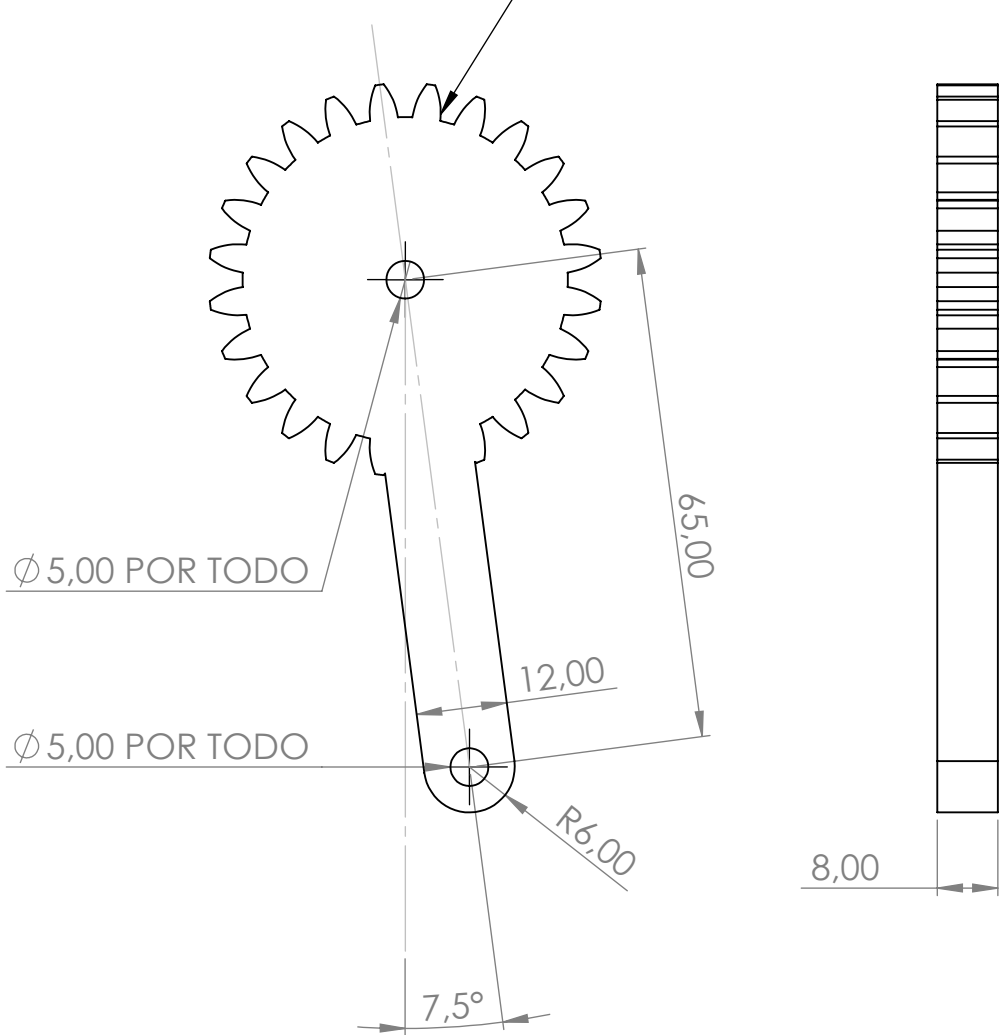
ESCALA:1:2

HOJA 1 DE 1

4 3 2 1



Características engranaje:
 Diámetro de paso=48mm
 Número de dientes =24
 Módulo=2



SI NO SE INDICA LO CONTRARIO:
 LAS COTAS SE EXPRESAN EN MM

TFM: Diseño y fabricación de una herramienta multifunción para un robot industrial

Master Ingeniería industrial
 Universidad de Almería

AUTOR: BORJA HIERRO GARRIDO



PESO: 14g

DESCRIPCIÓN: Eslabon con engranaje para entrada de potencia al sistema de pinza

MATERIAL:
PLA

N.º DE DIBUJO
 Eslabon con engranaje modificado

A4

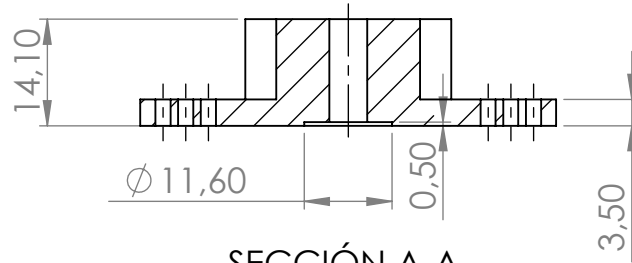
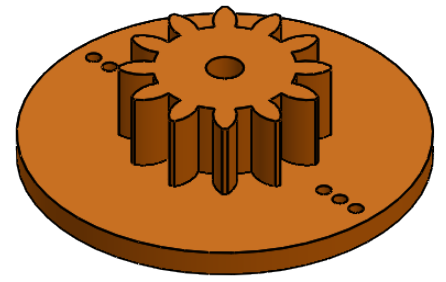
ESCALA: 1:1

HOJA 1 DE 1

4 3 2 1

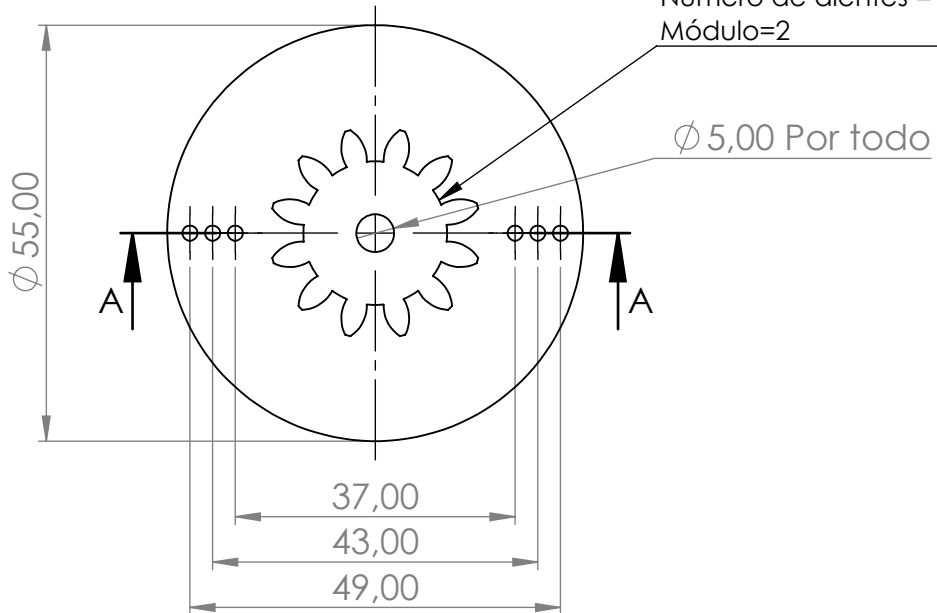
F

F



SECCIÓN A-A

Características engranaje:
 Diámetro de paso=24mm
 Número de dientes =12
 Módulo=2



NOTA: Todos los taladros $\phi 2$ Por todo, salvo indicación

SI NO SE INDICA LO CONTRARIO:
 LAS COTAS SE EXPRESAN EN MM

TFM: Diseño y fabricación de una herramienta multifunción para un robot industrial

Master Ingeniería industrial
 Universidad de Almería

AUTOR: BORJA HIERRO GARRIDO



PESO: 12g

DESCRIPCIÓN: Pieza de acoplamiento para transferir la potencia del servomotor a la pinza

MATERIAL:
PLA

N.º DE DIBUJO
 Engranaje de acople al servomotor

A4

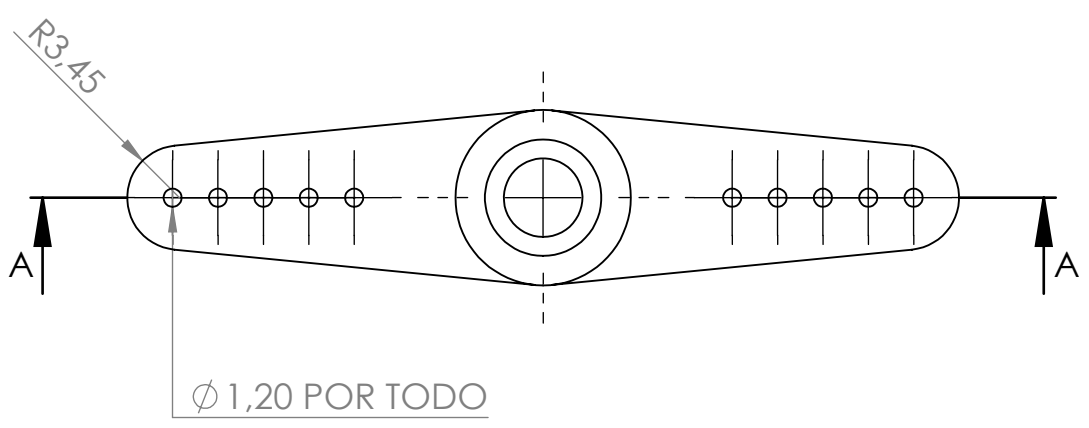
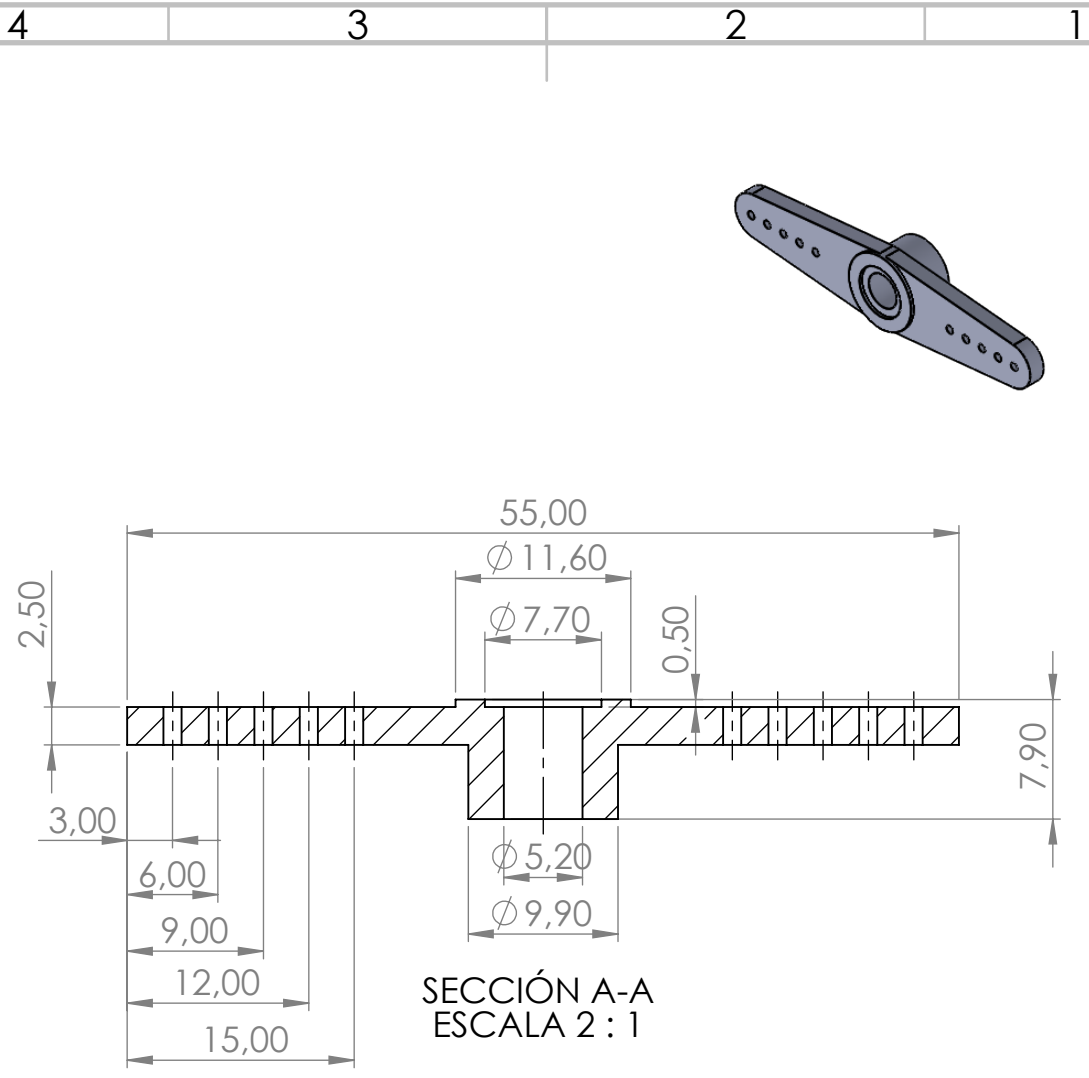
ESCALA:1:1

HOJA 1 DE 1

4 3 2 1

A

A



SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM

TFM: Diseño y fabricación de una herramienta multifunción para un robot industrial

Master Ingeniería industrial
Universidad de Almería

AUTOR: BORJA HIERRO GARRIDO



PESO: 2g

DESCRIPCIÓN: Elemento del servomotor para salida de giro

MATERIAL:
Comercial

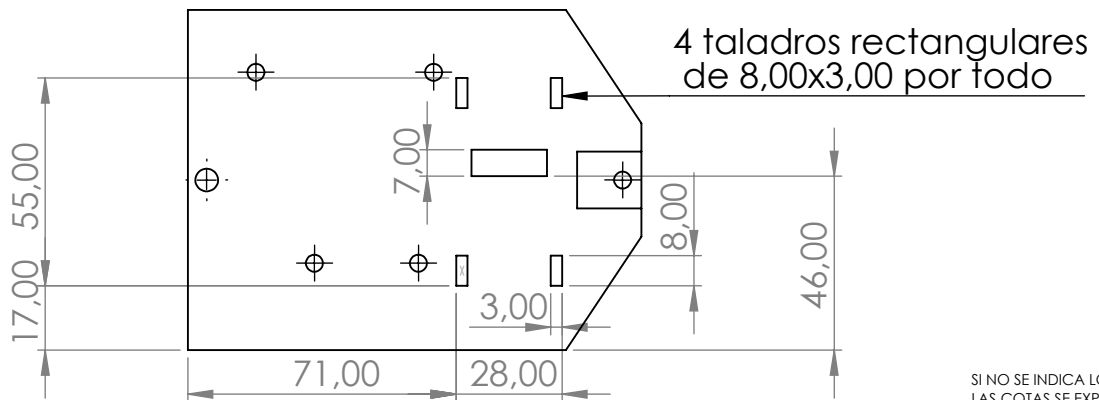
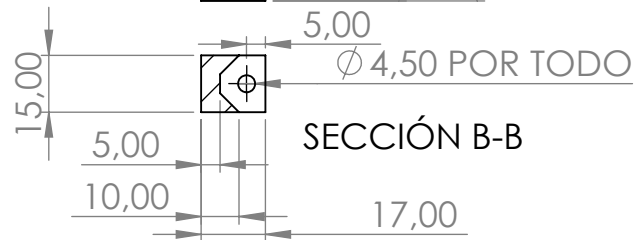
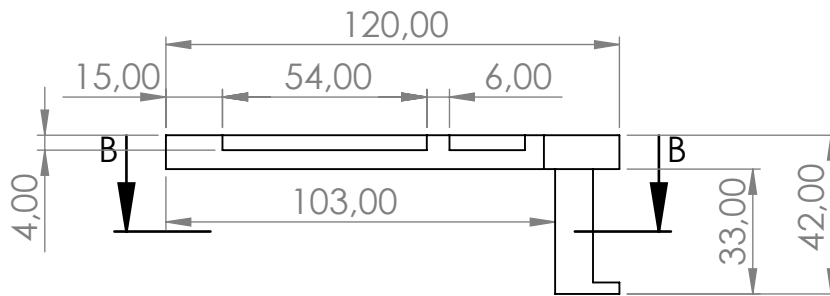
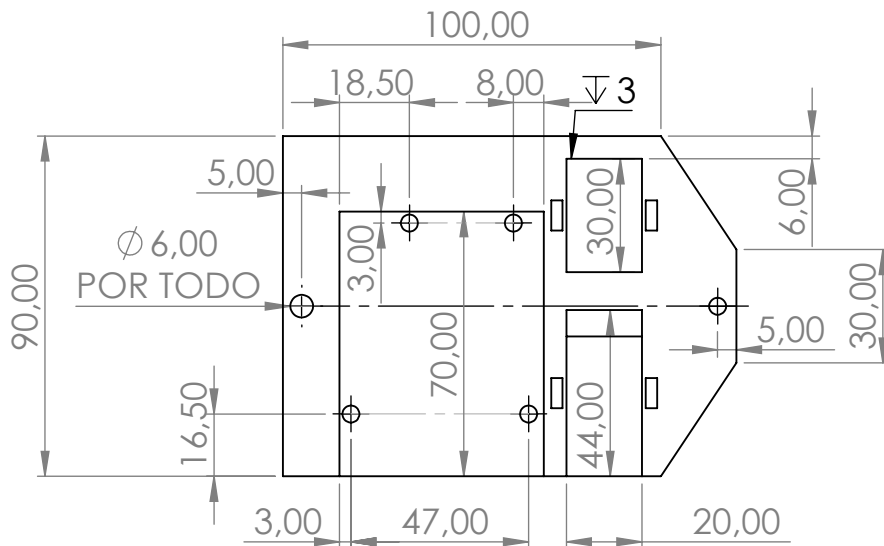
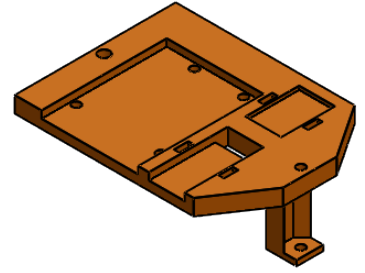
N.º DE DIBUJO
Acople servo

A4

ESCALA:2:1

HOJA 1 DE 1

Taladros de $\varnothing 4,50$ si no se indica lo contrario



SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM

TFM: Diseño y fabricación de una herramienta multifunción para un robot industrial

Master Ingeniería industrial
Universidad de Almería

AUTOR: BORJA HIERRO GARRIDO



PESO: 58g

DESCRIPCIÓN: Soporte situado en la parte superior de la herramienta

MATERIAL:
PLA

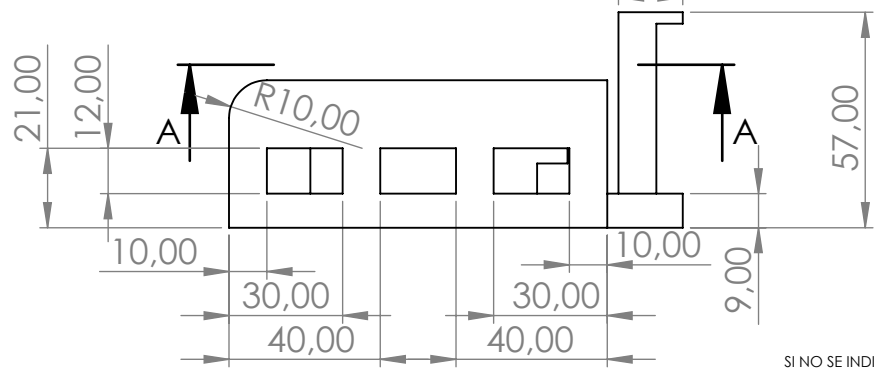
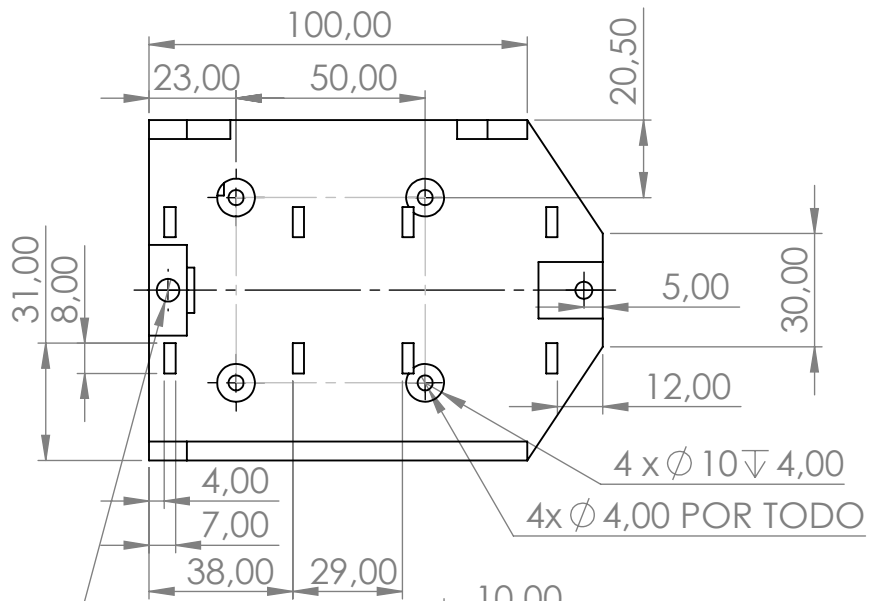
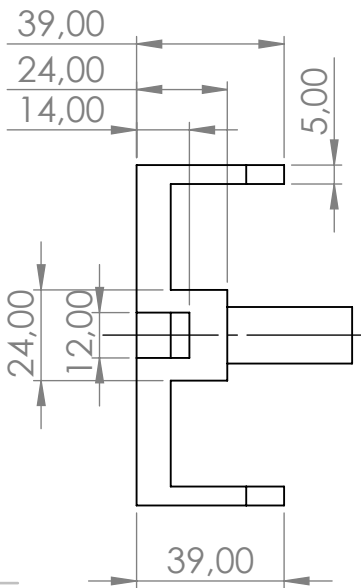
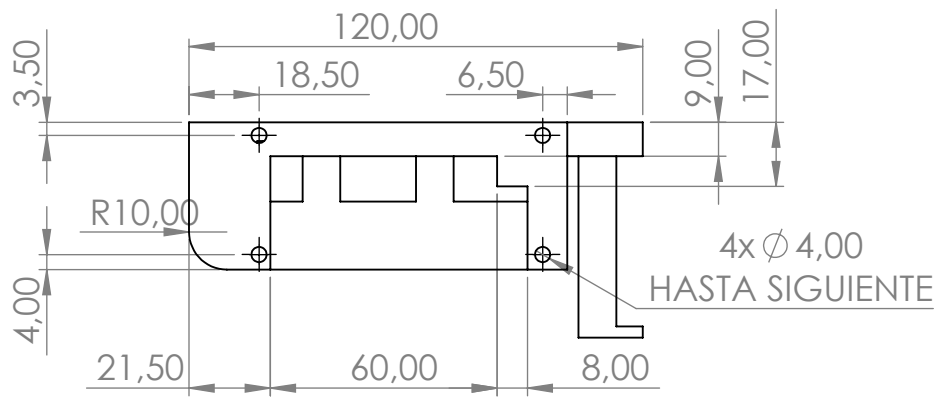
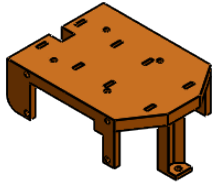
N.º DE DIBUJO

Soporte superior

A4

ESCALA:1:2

HOJA 1 DE 1



SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM

TFM: Diseño y fabricación de una herramienta multifunción para un robot industrial

Master Ingeniería industrial
Universidad de Almería

AUTOR: BORJA HIERRO GARRIDO



PESO: 77g

DESCRIPCIÓN: Soporte situado en la parte inferior de la herramienta

MATERIAL:
PLA

N.º DE DIBUJO
Soporte inferior

A4

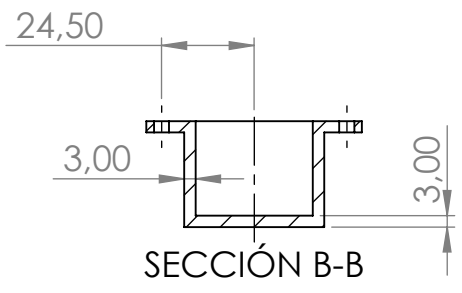
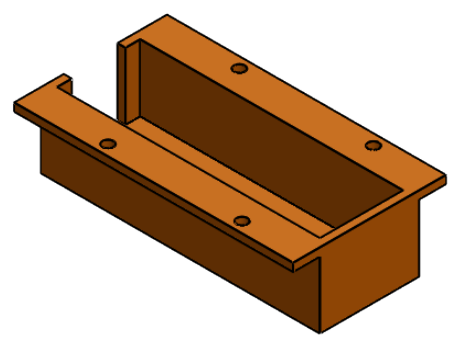
ESCALA:1:2

HOJA 1 DE 1

4 3 2 1

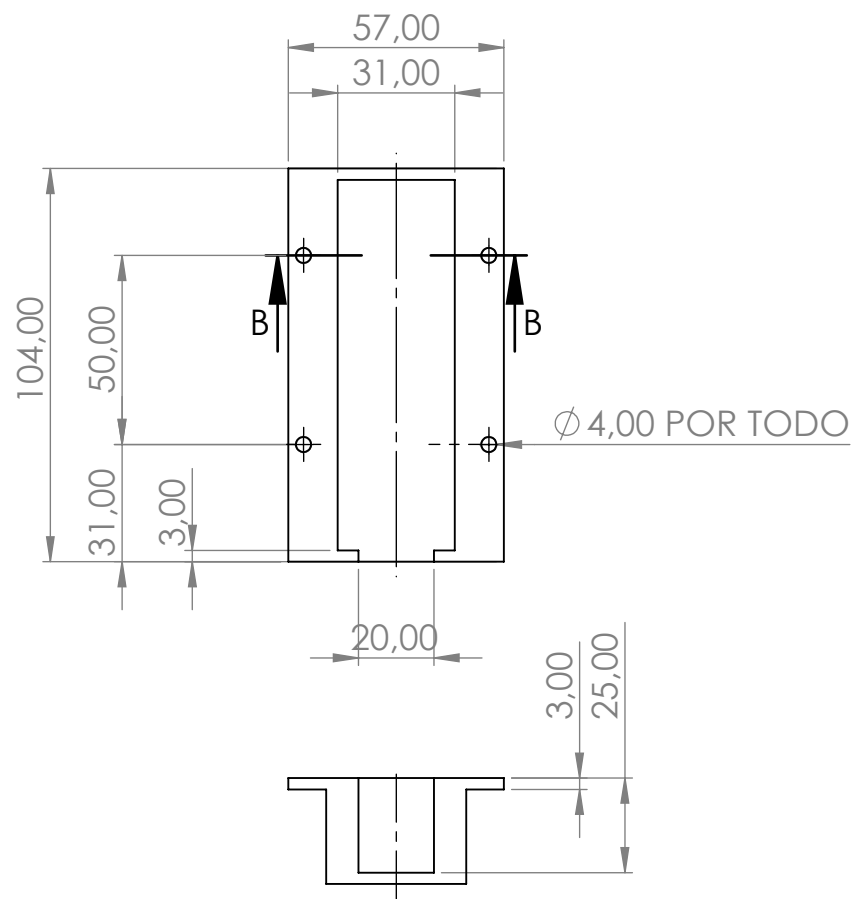
F

F



E

E



D

D

C

C

B

B

SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM

TFM: Diseño y fabricación de una herramienta multifunción para un robot industrial

Master Ingeniería industrial
Universidad de Almería

AUTOR: BORJA HIERRO GARRIDO



PESO: 52g

DESCRIPCIÓN: Soporte para acoplar la batería que alimenta al sistema

MATERIAL:
PLA

N.º DE DIBUJO
Soporte alimentación

A4

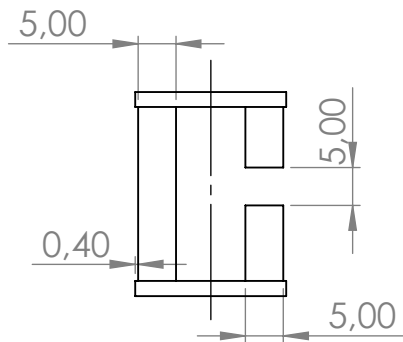
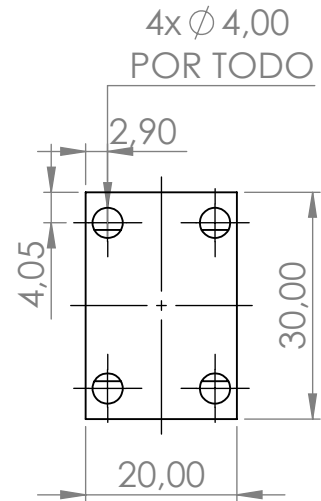
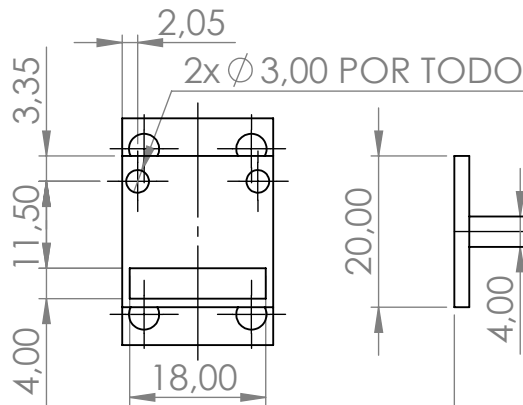
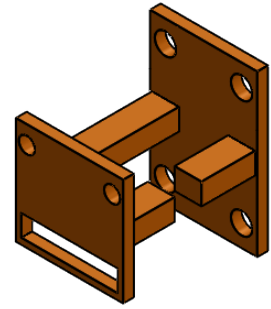
ESCALA:1:2

HOJA 1 DE 1

4 3 2 1

A

A



SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM

TFM: Diseño y fabricación de una herramienta multifunción para un robot industrial

Master Ingeniería industrial
Universidad de Almería

AUTOR: BORJA HIERRO GARRIDO



PESO: 5g

DESCRIPCIÓN: Soporte para el sensor de distancia

MATERIAL:
PLA

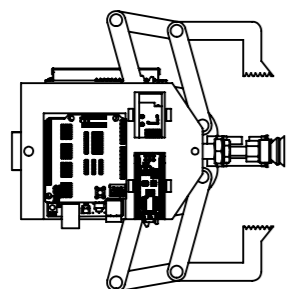
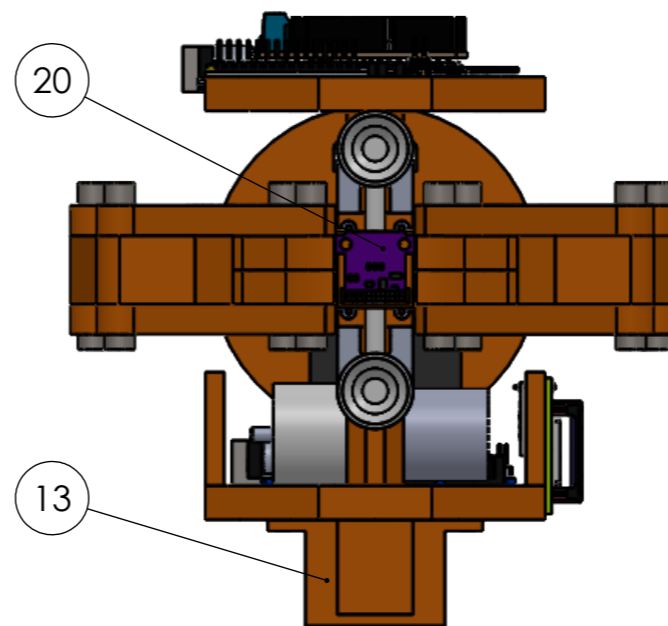
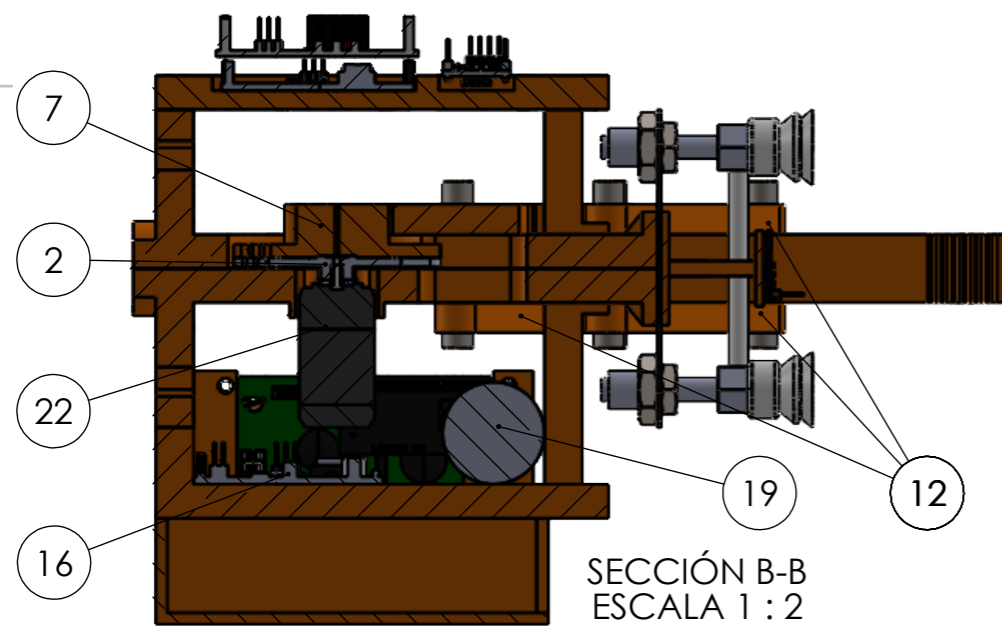
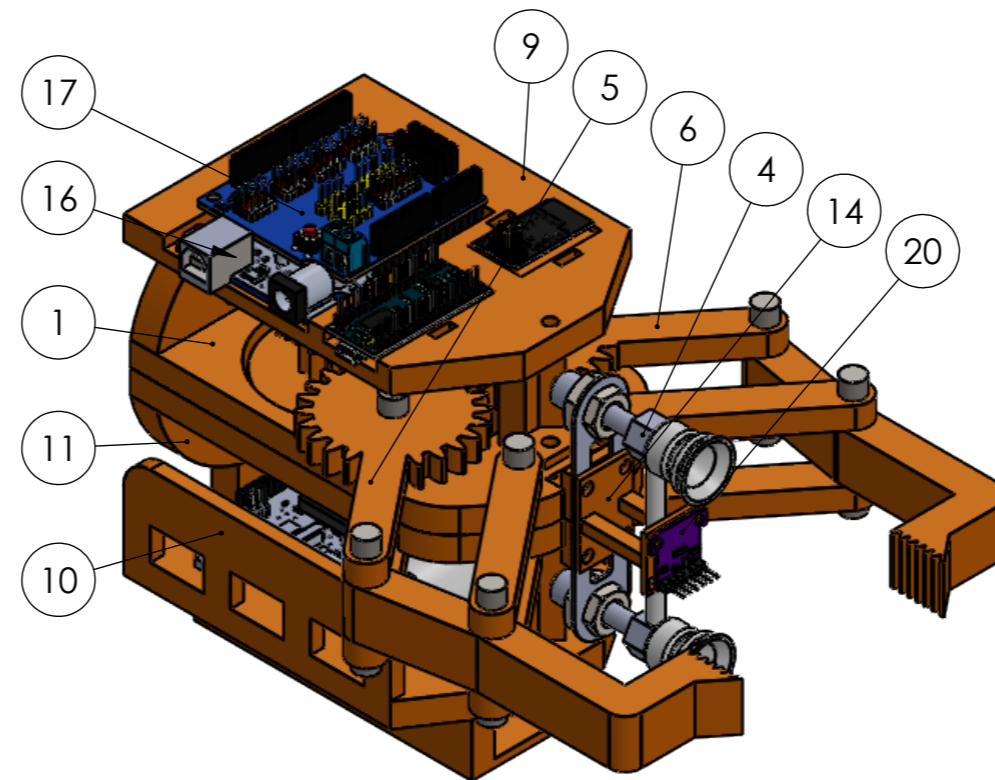
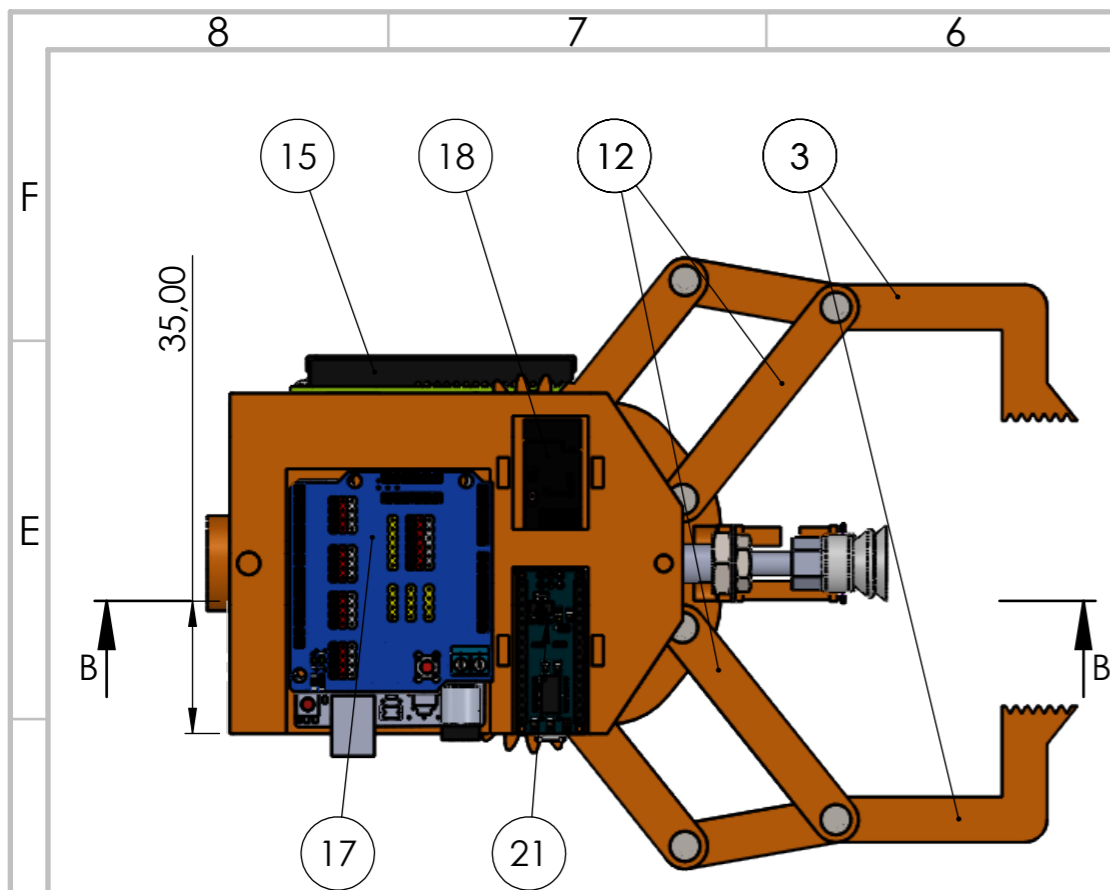
N.º DE DIBUJO

Soporte sensor

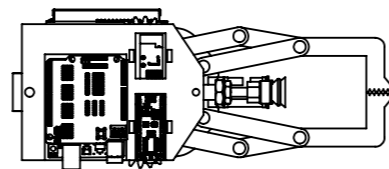
A4

ESCALA:1:1

HOJA 1 DE 1



APERTURA MÁXIMA (SERVO 180°)



CIERRE MÁXIMO (SERVO 0°)

N.º DE ELEMENTO	N.º DE PIEZA	CANTIDAD
1	BASE superior	1
2	Acople servo	1
3	Eslabón pinza	2
4	ENSAMBLAJE VENTOSAS	1
5	Eslabon con engranaje modificado	1
6	Eslabón con engranaje	1
7	Engranaje de acople al servo	1
9	Soporte superior	1
10	Soporte inferior	1
11	BASE inferior	1
12	Eslabon simple	6
13	CajaBateria	1
14	Soporte sensor diatancia	1
15	LCD AP214	1
16	Arduino Uno	2
17	Arduino Sensor shield v5.0	1
18	nRF24L01	1
19	Bomba_ventosa	1
20	Sensor distancia vI5310x	1
21	arduino nano	1
22	Servo Motor	1

SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM



PESO:
927g

MATERIAL:
PLA

UNIVERSIDAD DE ALMERÍA
MASTER EN INGENIERÍA INDUSTRIAL

TÍTULO:
Ensamblaje herramienta

AUTOR:
BORJA HIERRO GARRIDO

A3



DOCUMENTO 4: MEDICIONES Y PRESUPUESTO



Diseño y fabricación de una herramienta multifunción para un robot industrial.

1. PRESUPUESTO

Para poder obtener un presupuesto de la fabricación de esta herramienta multifunción se deberá distinguir entre los componentes que serán fabricados a través de impresión 3D y los elementos comerciales acoplados en ella.

También deberán tenerse en cuenta los costes asociados a la amortización de las herramientas utilizadas, su tiempo de uso y los costes de ingeniería asociados al desarrollo del proyecto.

1.1. COMPONENTES DE IMPRESIÓN 3D

El coste de los elementos fabricados a través de impresión 3D se calculará teniendo en cuenta el precio del material y la cantidad de material utilizado.

Tras la realización del proceso de fabricación de las piezas mediante impresión 3D se obtuvo la tabla correspondiente al tiempo y peso de cada una de ellas. [Tabla1]

IMPRESIÓN 3D					
	Pieza	t impresión	t [min]	Unidades	peso
1	BASE superior	4h 44 min	284	1	54g
2	BASE inferior	5h 38min	338	1	64g
3	Eslabón simple	30 min	30	6	6g
4	Eslabón pinza	1h 28 min	88	2	17g
5	Acople servo	1h 6min	66	1	12g
6	Eslabón con engranaje 1	1h 10 min	70	1	14g
7	Eslabón con engranaje 1	1h 10 min	70	1	14g
8	Soporte superior	5h 13min	313	1	58g
9	Soporte Inferior	7h 7min	427	1	77g
10	Caja batería	4h 37min	277	1	52g
11	Soporte Sensor distancia	30min	30	1	5g
TIEMPO TOTAL		33h 13min	1993		

Tabla 1. Tiempos y peso componentes 3D

Haciendo uso de esta tabla se podrá calcular el precio correspondiente a cada una de las piezas. [Tabla2]

Se ha tomado como precio representativo del kg de PLA el valor del filamento que se ha comprado para la realización de este proyecto:

$$\text{Precio PLA} = 19,99\text{€/kg} \approx 20\text{€/kg} = 0,02\text{€/G}$$

PIEZA	UNIDADES	PESO[G]	precio [€/g]	IMPORTE [€]
BASE superior	1	54	0,02	1,08
BASE inferior	1	64	0,02	1,28
Eslabón simple	6	6	0,02	0,72
Eslabón pinza	2	17	0,02	0,68
Acople servo	1	12	0,02	0,24
Eslabón con engranaje 1	1	14	0,02	0,28
Eslabón con engranaje 1	1	14	0,02	0,28
Soporte superior	1	58	0,02	1,16
Soporte Inferior	1	77	0,02	1,54
Caja batería	1	52	0,02	1,04
Soporte Sensor distancia	1	5	0,02	0,1
			TOTAL [€]	8,4

Tabla 2. Importe piezas individuales

El coste correspondiente al tiempo de utilización de la impresora y su amortización se calcularán de la siguiente manera: [Tabla3]

Concepto	CANTIDAD	PRECIO [€]	IMPORTE [€]
Amortización impresora	1	20	20
Tiempo máquina [h]	33,25	0,5	16,625
		TOTAL [€]	36,625

Tabla 3. Costes asociados impresión 3D

Se ha tenido en cuenta un precio de 20€ relativo a la amortización de la impresora, en el que se ha tenido en cuenta el precio inicial que supuso y el tiempo transcurrido desde su compra.

Para el precio fijado a la hora de utilización de la máquina se ha tenido en cuenta el suministro eléctrico directo de la impresora y derivado del uso de las instalaciones de la Universidad de Almería.

El precio total correspondiente a los elementos impresos mediante tecnología de impresión 3D es de 45,03€

1.2. COMPONENTES COMERCIALES

Se han reunido todos los componentes comerciales necesarios para completar el diseño de la herramienta y se le ha aplicado el precio de cada uno de ellos para obtener el presupuesto correspondiente [Tabla4]:

COMPONENTE	UNIDADES	Precio/Ud [€]	IMPORTE [€]
Arduino Uno	2	9,99	19,98
Arduino Nano	2	6,99	13,98
Sensor shield v5.0	1	2,89	2,89
Módulo de comunicación NRF24L01	2	2,69	5,38
Servomotor. [Modelo Power HD; HD-3001HB]	1	9,95	9,95
Sistema de ventosas con bomba de succión. [Kit para 25T Servos MG996 MG995 DS3218]	1	15,96	15,96
Pantalla Lcd y módulo de comunicación I2C	1	6,99	6,99
Sensor de distancia VL53L0X	1	7,69	7,69
ProtoBoard	1	36,49	36,49
Interruptores	6	1,92	11,52
Led	6	0,6	3,6
Alimentación (Sistema PowerBank)	1	11,1	11,1
Cableado de conexión	PACK	6,99	6,99
		TOTAL [€]	152,52

Tabla 4. Importes componentes comerciales

El precio total correspondiente a los elementos comerciales es de 152,52€

1.3. COSTES DE INGENIERÍA

Por último, se añadirán los costes de mano de obra relacionados con la ingeniería que ha tenido lugar detrás de este proyecto y el de amortización de los equipos tales como ordenador y software.

Al tratarse de un proyecto que se ha desempeñado dentro de una labor docente y de estudio, el coste de la hora de ingeniería no será lo correspondiente al precio que supondría si este proyecto se realizase dentro de una empresa del sector privado, si no que el coste se ha estimado en 20€/h, bastante por debajo del mercado dentro del ámbito laboral habitual. [Tabla5]

CONCEPTO	CANTIDAD	PRECIO[€/h]	IMPORTE
Mano de obra ingeniero [h]	385	20	7700
Amortización equipos informáticos y licencias de Software	1	100	100
		TOTAL [€]	7800

Tabla 5. Costes de ingeniería

El precio total asociado a los costes de ingeniería es de 7800€

1.4. PRESUPUESTO TOTAL

Teniendo en cuenta los tres conceptos anteriormente desarrollados, podremos calcular nuestro presupuesto total de ejecución del proyecto [Tabla6]:

CONCEPTO	IMPORTE [€]
Componentes impresión 3D	45,03
Componentes comerciales	152,52
Coste de ingeniería	7800
TOTAL PRESUPUESTO EJECUCIÓN	7997,55

Tabla 6.Presupuesto de ejecución

Para alcanzar el presupuesto total se ha tenido en cuenta un margen de beneficio del 7,5%, así como el IVA correspondiente. [Tabla7]

CONCEPTO	[%]	IMPORTE [€]
PRESUPUESTO DE EJECUCIÓN		7997,55
%BENEFICIO	0,075	599,81625
%IVA	0,21	125,9614125
TOTAL PRESUPUESTO		8723,327663

Tabla 7.Presupuesto final

Finalmente se obtiene un presupuesto total de OCHO MIL SETECIENTOS VEINTITRÉS EUROS CON TREINTA Y CUATRO CÉNTIMOS.

En este trabajo se detalla el proceso de diseño y fabricación de una herramienta multifunción orientada a su uso mediante brazo robótico.

En la actualidad existe una clara tendencia hacia la industria 4.0 y la implementación de robots dentro de los procesos industriales. Este TFM se centra en desarrollar una herramienta completamente funcional orientada a ofrecer soluciones para la manipulación de elementos a través del uso de los robots industriales.

A lo largo de este proyecto se seleccionarán las soluciones más satisfactorias para conseguir los objetivos buscados. Se realizarán los diseños y comprobaciones necesarias para garantizar la calidad y el correcto dimensionamiento de la herramienta. Adicionalmente se realizarán las simulaciones y ensayos reales para verificar su correcto funcionamiento dentro del sistema que incluye la unión de la herramienta junto con el robot industrial y su sistema de control.

This work provides a detail explanation of the fabrication and design process of a multifunctional tool oriented to its use by means of a robotic arm.

Currently there is a clear trend towards Industry 4.0 and the implementation of robots within industrial processes. This project focuses on developing a fully functional tool aimed at offering solutions for the manipulation of elements through the use of industrial robots.

Throughout this project the most satisfactory solutions will be selected to achieve the desired objectives. The designs and verifications necessary to guarantee the quality and correct sizing of the tool will be done. Additionally, real simulations and tests will be carried out to verify its correct operation within the system that includes the union of the tool together with the industrial robot and its control system.