



Diseño, fabricación, montaje y test de un robot tipo SCARA

Pilar Estefanía Samaniego Villacres pilar.samaniego@eu4m.eu

Marta Casasola Paesa UO237179@uniovi.es

Manuel Alberto Martínez Arenas manuel.martinez@eu4m.eu

José Eduardo Villanueva Zacarías eduardo.villanueva@eu4m.eu

Abstract— El presente proyecto consiste en diseñar, fabricar, montar y testear un robot tipo SCARA con 4 grados de libertad. El Tool Center Point (TCP) podrá describir trayectorias tridimensionales predefinidas por el usuario y además podrá rotar respecto al eje vertical (eje Z), será controlado mediante 4 motores DC y su programación será realizada en Arduino.

Palabras claves— Arduino, CAD-CAE, cinemática inversa, controlador, Denavit–Hartenberg, diseño basado en modelo, Matlab, prototipo, SCARA, Shield, simulación dinámica, Simulink, síntesis, TCP

I. INTRODUCCIÓN

El modelado, la simulación y el análisis de un robot previo a su fabricación y montaje es una parte esencial en el diseño y construcción del mecanismo, ya que permite una verificación anticipada del mismo. Además, permite realizar pruebas con el fin de comprobar qué características del diseño son optimizables de forma sencilla y relativamente rápida en comparación con un prototipo físico.

El robot SCARA a diseñar se clasifica en la categoría tipo C de 1ª generación de la Federación Internacional de Robótica. Esta categoría lo clasifica como un robot programable con trayectoria continua o punto a punto sin realimentación.

Es un robot tipo serie, es decir, se caracteriza por la cadena abierta que forman sus eslabones y pares cinemáticos. El espacio de trabajo que estos robots ofrecen es amplio. Sin embargo, poseen una baja relación masa/rigidez y, por tanto, con el fin de asegurar una determinada rigidez, sus eslabones han de ser robustos. Es por ello por lo que las masas que son capaces de mover y las aceleraciones no son muy elevadas.

II. PLANIFICACIÓN Y COORDINACIÓN DEL PROYECTO

Desde el arranque del proyecto se realizaron una serie de reuniones de trabajo para definir los objetivos intermedios que se deberían ir cumpliendo, así como las fechas de entrega de dichos avances y para apoyarnos en la organización se utilizaron una serie de herramientas informáticas de uso libre como es el software Teams de Microsoft, la plataforma de para administración de proyectos Trello y la herramienta gráfica de seguimiento BigPicture.

En los anexos se puede observar el mapa de Gantt donde se representa de manera gráfica el desarrollo de este proyecto, desde la recopilación de información existente hasta la entrega del prototipo funcional del robot SCARA.

III. ESPECIFICACIONES TÉCNICAS

La estructura del robot propuesta es la mostrada en Fig. 1. Este robot SCARA tiene 4 grados de libertad (GDL) e incluye 3 pares de revolución y un par prismático. Tal como se muestra en la misma figura, los ejes X e Y se encuentran en el plano horizontal de la base mientras que el eje Z se ubica en posición vertical.

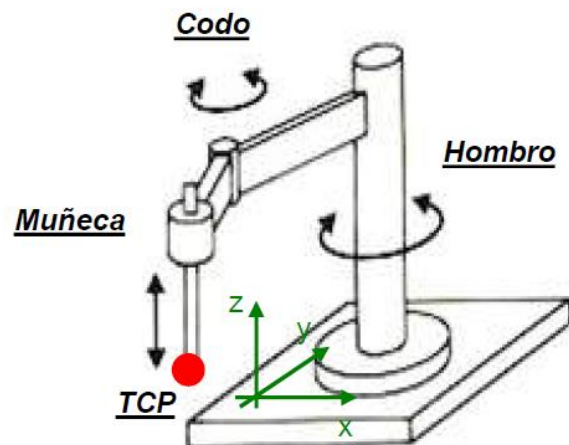


Fig. 1. Estructura de robot SCARA propuesta.

El espacio de trabajo para el cual el robot ha de ser diseñado debe ser simétrico y tener las siguientes dimensiones:

$$x_{TCP} \in [0,150] \text{ mm}$$

$$y_{TCP} \in [0,100] \text{ mm}$$

$$z_{TCP} \in [0,150] \text{ mm}$$

Además, se requiere un ángulo de transmisión en el codo comprendido en el rango $[60, 120]^\circ$ y un giro de la muñeca de 360° . En el TCP debe incluirse una pinza que admita los modos abierto o cerrado.

El modo de funcionamiento del robot podrá ser manual o automático. En el primero, los actuadores serán controlados individualmente desde un ordenador al cual estará conectado el



robot. En el segundo, el TCP deberá seguir, previa programación, una trayectoria mientras la muñeca efectúa el giro.

A. Síntesis dimensional

Mediante la resolución del problema de síntesis dimensional se obtuvieron las medidas del brazo y del antebrazo para el espacio de trabajo especificado. En el robot tipo Scara, ésta se ve simplificada dado que el movimiento vertical puede desacoplarse del resto de eslabones del brazo, los cuales, junto a las rotaciones del hombro, codo y muñeca, pueden tratarse como un mecanismo plano.

Dos configuraciones se muestran en la Fig. 2: la configuración 1 es el caso del límite inferior del ángulo de transmisión del codo (μ_{\min}), mientras que la configuración 2 es el límite superior (μ_{\max}).

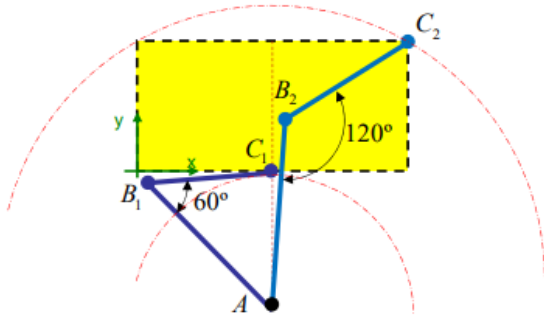


Fig. 2. Configuraciones y espacio de trabajo.

Los vectores de constantes, \mathbf{p} , e incógnitas, \mathbf{b} , serán, respectivamente:

$$\mathbf{p} = [x_a \ \mu_{\min} \ \mu_{\max} \ x_{c_1} \ y_{c_1} \ x_{c_2} \ y_{c_2}] \quad (1)$$

$$\mathbf{b} = [y_a \ x_{b_1} \ y_{b_1} \ x_{b_2} \ y_{b_2} \ Lab \ Lbc]^T \quad (2)$$

Se plantean las siguientes ecuaciones de la cinemática:

$$(xb_1 - xa)^2 + (yb_1 - ya)^2 - Lab^2 = 0 \quad (3)$$

$$(xc_1 - xb_1)^2 + (yc_1 - yb_1)^2 - Lbc^2 = 0 \quad (4)$$

$$(xa - xb_1) \cdot (xc_1 - xb_1) + (ya - yb_1) \cdot (yc_1 - yb_1) - Lab \cdot Lbc \cdot \cos(\mu_{\min}) = 0 \quad (5)$$

$$(xb_2 - xa)^2 + (yb_2 - ya)^2 - Lab^2 = 0 \quad (6)$$

$$(xc_2 - xb_2)^2 + (yc_2 - yb_2)^2 - Lbc^2 = 0 \quad (7)$$

$$(xa - xb_2) \cdot (xc_2 - xb_2) + (ya - yb_2) \cdot (yc_2 - yb_2) - Lab \cdot Lbc \cdot \cos(\mu_{\max}) = 0 \quad (8)$$

La resolución de este sistema se realizará mediante el método iterativo de Newton, pero ya que la matriz jacobiano no es

cuadrada -el sistema tiene 6 ecuaciones y 7 incógnitas, es decir, es infradeterminado- es necesario utilizar la matriz pseudoinversa, J^+ . Al ser un sistema infradeterminado y, por tanto, tener infinitas soluciones, la solución que se obtenga con (3) será la más parecida a la solución inicial aproximada.

$$b_{i+1} = b_i - J^+ \cdot \Omega \quad (9)$$

En dicha ecuación Ω representa el sistema de ecuaciones. Tras 7 iteraciones y una tolerancia del error menor a 10^{-8} se obtienen los valores redondeados en la TABLA I. (en mm), que son las dimensiones principales para el diseño del mecanismo y que definen la cinemática.

TABLA I. DIMENSIONES PRINCIPALES

Parámetro	Valor
y_a	-171
Lab	195
Lbc	128

B. Simulación y control

Como ya se mencionó anteriormente, el manipulador Scara tiene 4 grados de libertad – las coordenadas x, y, z del TCP y δ , el giro del eslabón 4 respecto a la referencia absoluta. En las articulaciones A, B y C de la Fig. 3 están presentes actuadores de rotación para realizar el control de giro relativo entre los eslabones, mientras que para realizar el movimiento lineal del TCP (punto D) se emplea un actuador lineal.

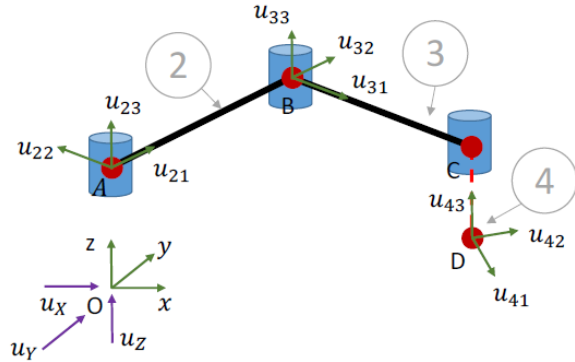


Fig. 3. Esquema del brazo robótico con articulaciones y eslabones.

Mediante la modelización cinemática, la formulación de matriz R y el integrador Runge-Kutta de 2º orden e incluyendo un control CTC, se obtienen los resultados de la simulación de la dinámica para el seguimiento de la trayectoria de la posición, velocidad y aceleración deseadas de las coordenadas independientes Fig. 4. En las gráficas inferiores se muestra el valor del error cometido de la posición.

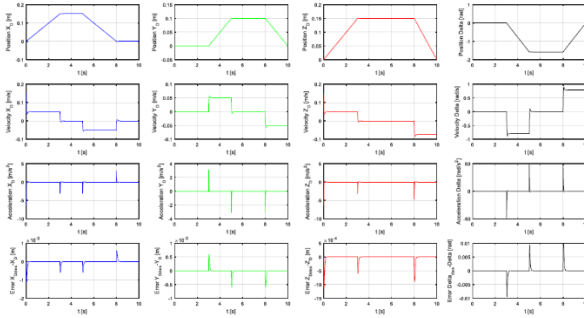


Fig. 4. Seguimiento de la trayectoria de la posición, velocidad y aceleración deseadas de las coordenadas independientes

Para conseguir este movimiento, se necesitan en los actuadores de las articulaciones los movimientos y fuerzas y pares que se muestran en la Fig. 5, los cuales se especifican en la TABLA I. .

TABLA II. RESULTADOS DE LA SIMULACIÓN

Actua- dor	Rango		
	Longitud/Giro	Velocidad	Fuerza/Par
Rota- ción A	[97.8 , 152.8]°	[-6.1 , 2.8] rpm	[-0.26 , 1.92] N·m
Rota- ción B	[-120.3 , - 60.1]°	[-6.8 , 6] rpm	[-0.54 , 0.64] N·m
Rota- ción C	[-158.4 , 7.6]°	[-13.1 , 12.6] rpm	[-0.05 , 0.05] N·m
Des- plaza- miento CD	[150 , 300] mm	[-150 , 85.5] mm/s	[-2.97 , - 1.04] N

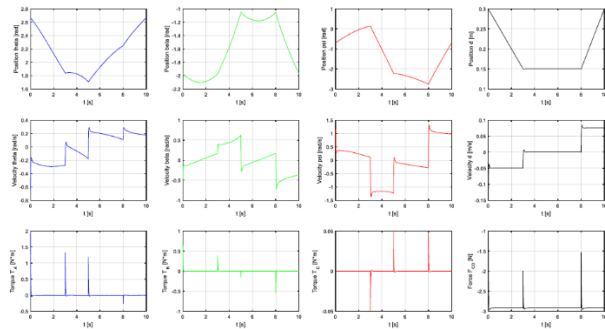


Fig. 5. Movimientos y fuerzas y pares de los actuadores en las articulaciones

Los motores finalmente seleccionados son:

- Articulación A – Motor de engranajes DC Micromotors E192-24-336
- Articulación B – 378:1 Metal Gearmotor 25Dx58L mm LP 12V
- Articulación C – 1000:1 Micro Metal Gearmotor HPCB 12V
- Movimiento lineal entre C y D - 1000:1 Micro Metal Gearmotor HPCB 12V

Las principales características se muestran en la TABLA III.

TABLA III. CARACTERÍSTICAS DE LOS TRES MOTORES DC

Motor	V [V]	Tmax [Nm]	Imax [A]	w[rpm]
BRAZO- E192.24.36	24	3	0.34	11.5
CODO 378:1 25Dx58L 12V	12	1.13	1.1	14
ALTURA Y PINZA 1000:1 - HPCB 12V	12	1	0.75	35

Una explicación más amplia sobre la síntesis dimensional y la simulación dinámica y el control pueden encontrarse en los anexos.

IV. PROCESO DE DISEÑO

En base a los resultados obtenidos de la síntesis dimensional, se inició un proceso iterativo de diseño, regido inicialmente por las siguientes premisas.

- Portabilidad: Contar con suficiente espacio para integrar los componentes electrónicos dentro del modelo y con ello transportar un único elemento
- Evitar la necesidad de un anclaje externo
- Geometría simple

De forma resumida se presenta una muestra de la evolución del diseño hasta llegar a la propuesta definitiva con la que se fabricó el prototipo.

A. Propuesta inicial

1. Eslabones

Tal como se puede observar en la Fig. 6, los eslabones móviles del robot fueron concebidos inicialmente como una estructura compuesta de dos placas rígidas unidas en cuatro puntos por tornillos para mantener una separación constante entre las mismas. La geometría de las placas se basó en círculos de diámetro descendente, conforme se avanza en la cadena cinemática, buscando con ello reducir la masa suspendida en el extremo del brazo y mantener cualitativamente una inercia elevada en el sentido de movimiento de los eslabones.

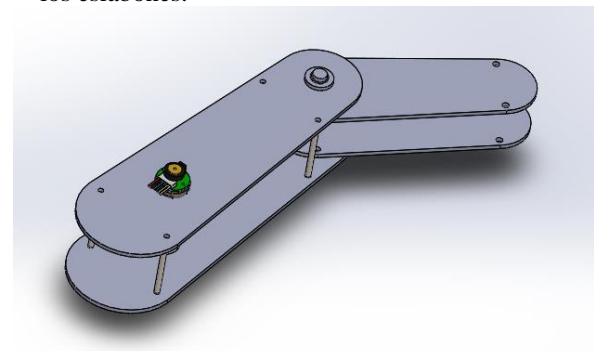


Fig. 6. Primera idea de perfil para los eslabones móviles

2. Base

Con amplias dimensiones a comparación de los demás elementos (200mm x 200mm x 130mm), se buscó generar un



espacio interior que permitiera instalar la placa de conexiones, la placa de desarrollo, el motor responsable del movimiento de la primera articulación y los elementos de interacción con el usuario (LCD para mostrar información sobre el sistema y controles manuales).

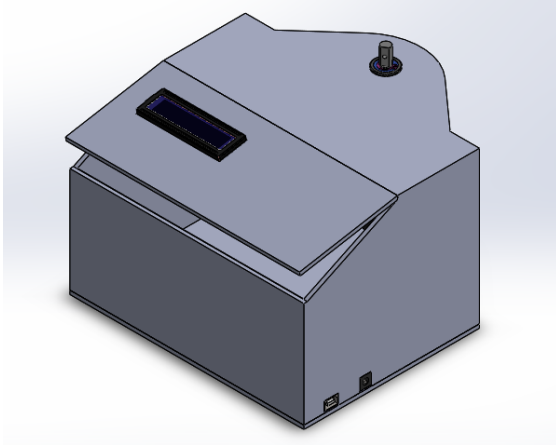


Fig. 7. Primera propuesta del eslabón base

Se pensó que la fabricación de este elemento fuera por medio de lámina de acero doblada, con la intención de reducir la complejidad del maquinado de las piezas y de igual manera se pensó que cumpliera dos funciones simultáneamente: por un lado, la función estructural para soportar las cargas generadas por los eslabones móviles y por otro la de protección de los elementos electrónicos para evitar incidentes imprevistos durante la manipulación del robot.

B. Primera revisión

Tras una primera revisión de la propuesta se identificaron ciertas áreas de oportunidad en el diseño por lo que se realizó una serie de cambios buscando soluciones alternativas que solventarán los requerimientos descritos.

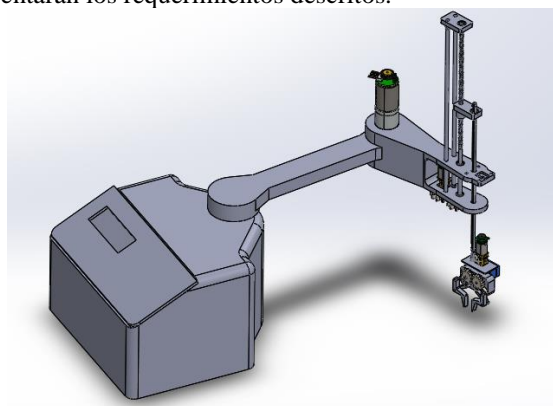


Fig. 8. Ensamble general tras la primera revisión

Así se llegó a la segunda versión del prototipo Fig. 8 con las siguientes características.

1. Eslabones

Se modificó la geometría de los eslabones por un perfil aligerado que constara en lo posible de una sola pieza para mejorar la rigidez de cada elemento independiente además de reducir la complejidad del montaje.

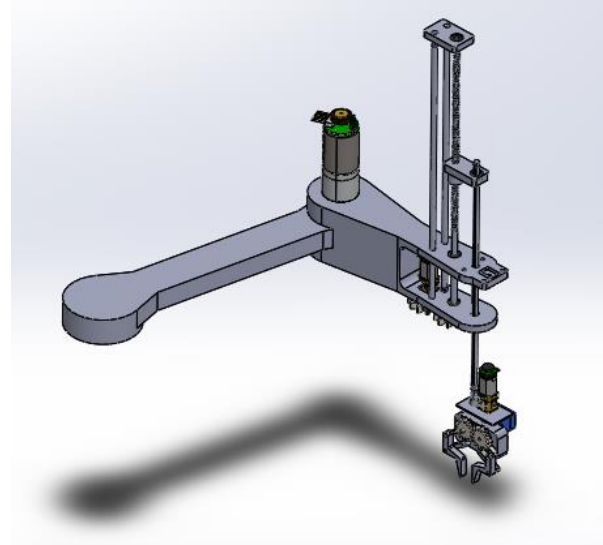


Fig. 9. Segunda propuesta de la geometría de los eslabones

Bajo el mismo principio de reducción del número de componentes se planteó la utilización de la propia geometría del eslabón como guía y/o soporte de elementos adicionales con los que interactúan, ya sea motores, guías, rodamientos, entre otros.

2. Junta prismática

Luego de analizar una serie de posibilidades, se decidió ubicar la junta prismática que posibilita el movimiento de elevación, en el extremo final del robot. En esta propuesta se planeó generar el desplazamiento vertical mediante la utilización de un husillo impulsado por un motor de poca potencia a través de una transmisión rígida de engranes.

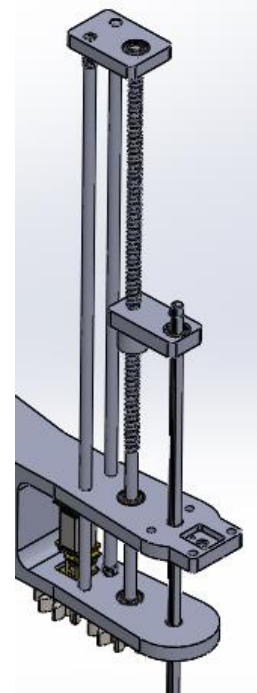


Fig. 10. Junta prismática con guía soportada por dos postes

Para prevenir la desalineación del husillo y con ello reducir las pérdidas mecánicas durante el movimiento de esta junta, se propuso la creación de una estructura de soporte para el

husillo, conformada por un par de tubos metálicos apoyados, cada uno, en dos puntos al segundo eslabón del robot y unidos al husillo mediante un pequeño elemento rígido. Fig.5

3. Actuador final

En esta versión del prototipo, se diseñó una pinza a partir de un doble mecanismo de cuatro barras, con un tercer eslabón modificado para hacer la función de “dedo”. El mecanismo se dispuso de forma simétrica, para que un único servomotor pudiera propulsar ambos cuadriláteros simultáneamente y con ello evitar problemas de sincronización.

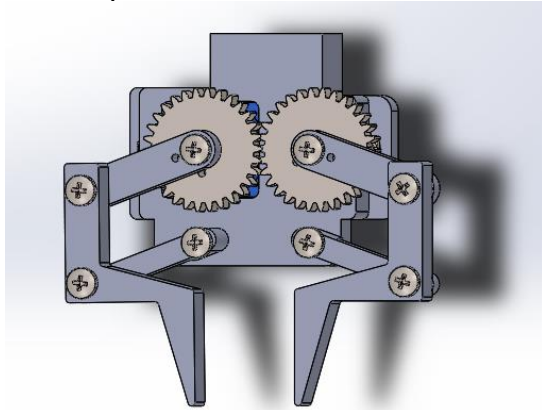


Fig. 11. . Pinza basada en doble mecanismo de cuatro barras

Una característica particular de este modelo es el mecanismo que se utilizó para ensamblar el actuador al resto del robot, el cual permite un reemplazo sencillo del elemento.

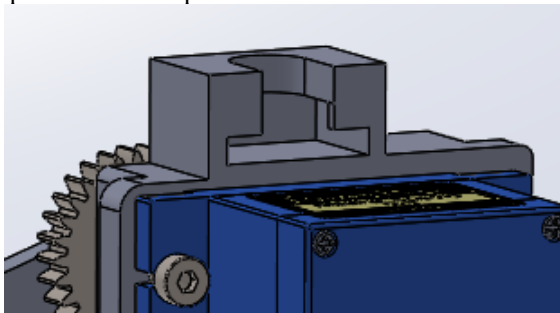


Fig. 12. . Cavity with defined geometry for rapid assembly

Se generó una cavidad de forma regular que permitiera introducir una pieza con la geometría descrita por el perfil de la cavidad y además se dio una profundidad ligeramente mayor a la cavidad para permitir que la pieza introducida alcance una posición de empotramiento y quede bloqueada en todas direcciones excepto una.

C. Segunda revisión

Como parte del refinamiento del modelo se solicitó la asesoría en cada una de las áreas involucradas (mecánica, fabricación y electrónica) y con base en sus observaciones se establecieron nuevas premisas para regir el proceso de diseño:

- Montaje modular
- Capacidad de ajuste: posibilidad de corregir desalineaciones de ejes, engranajes
- Aprovechamiento del material estandarizado y ya disponible.

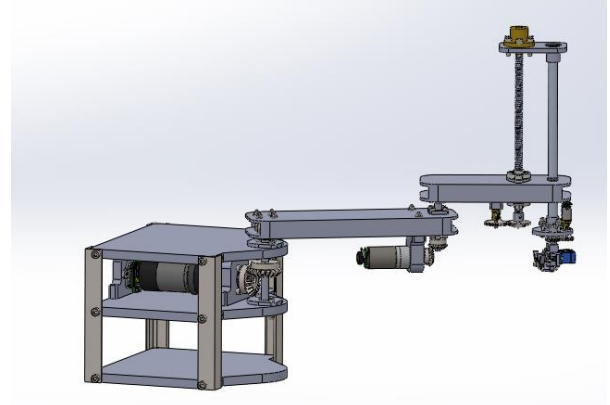


Fig. 13. Modelo completo tras la segunda revisión

1. Base

Tras haberse señalado tanto las deficiencias en términos estructurales como las dificultades en materia de fabricación, se generó una estructura completamente distinta a la versión anterior, teniendo como única similitud, la existencia de un amplio espacio interior donde colocar los componentes electrónicos.

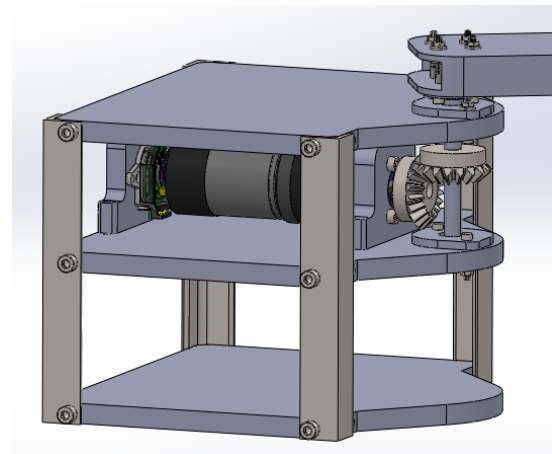


Fig. 14. Eslabón base conformado por tres placas y perfil L

La nueva estructura se concibió como el ensamble de tres placas metálicas en una disposición vertical, unidas entre sí mediante cuatro perfiles L fijados a las caras laterales de las placas mediante tornillos.

Debido a la nueva estructura, se modificó igualmente la transmisión del motor responsable de la rotación del primer eslabón. Se eligió una transmisión rígida de engranes cónicos, con lo que se logró cambiar la posición de montaje del motor, maximizando el área disponible para la placa electrónica.

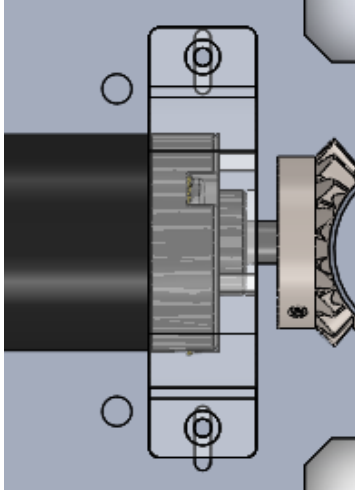


Fig. 15. Ranuras rectas para ajuste fino en la alineación

De acuerdo con las nuevas condiciones de diseño, se cambiaron los taladros de fijación para la base del motor por ranuras rectas haciendo posible un ajuste fino en la alineación de los ejes.

2. Eslabones

Conservando la idea de generar el mayor momento de inercia en la dirección del movimiento, se optó por utilizar un perfil rectangular estándar de aluminio de 70x20 mm.

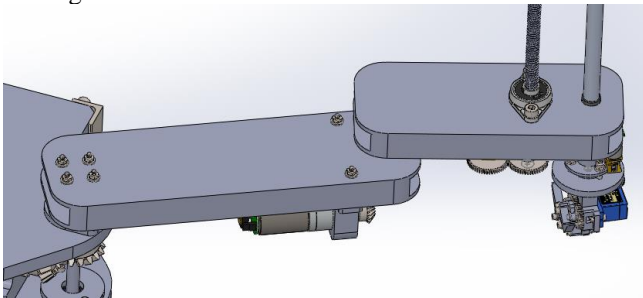


Fig. 16. Eslabones móviles utilizando perfil estándar

Al igual que en la primera articulación, la transmisión seleccionada para propulsar el segundo eslabón móvil es de tipo rígida con engranes cónicos.

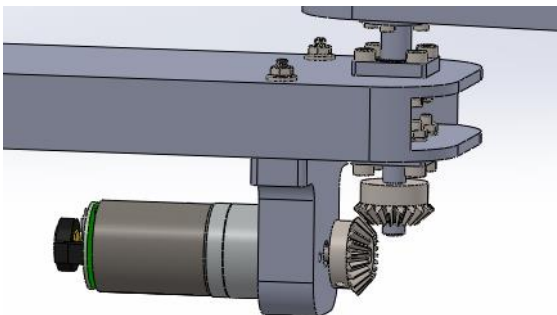


Fig. 17. Transmisión rígida de engranes cónicos entre eslabones móviles

3. Junta prismática

Considerando la escala del modelo completo y el orden de las fuerzas presentes, se determinó que la rigidez propia del hu-

sillo espiral de 8mm apoyado en ambas caras del eslabón sería suficiente para evitar desalineaciones problemáticas del elemento deslizante.

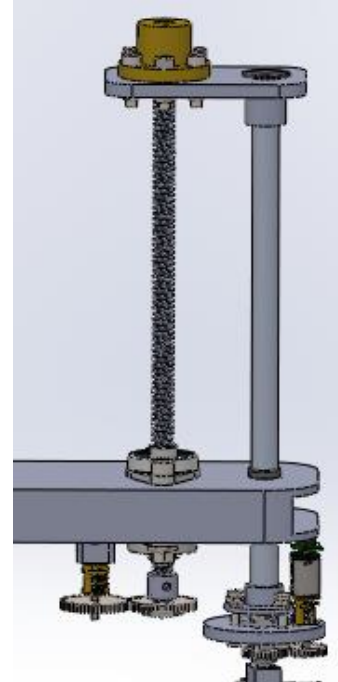


Fig. 18. Junta prismática sin soportes adicionales

D. Modelo Definitivo

Habiendo satisfecho los requisitos mencionados con anterioridad, se solicitaron de nueva cuenta recomendaciones respecto a la viabilidad de manufactura de los elementos y se realizó una serie de modificaciones acorde con los comentarios recibidos.

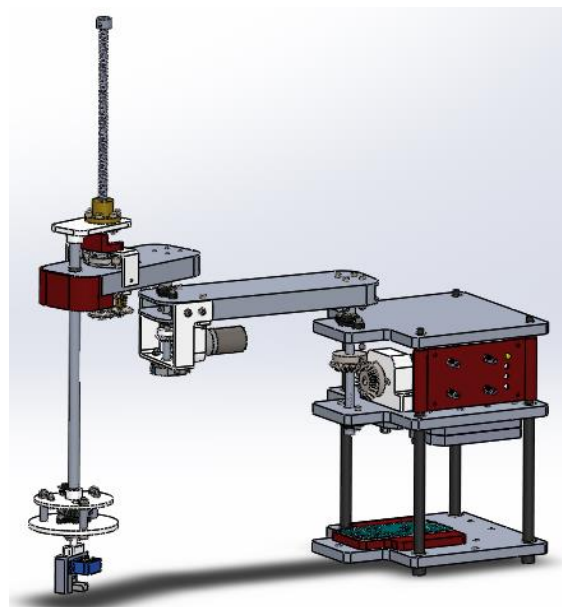


Fig. 19. Modelo definitivo del prototipo

1. Base

Se eliminaron los taladros roscados en las caras laterales de las placas que conforman la base del robot, sustituyéndolos por cuatro taladros pasantes en cada una de las tres piezas. En

consecuencia, la forma de ensamblar la estructura es mediante un montaje vertical de las placas sobre cuatro postes metálicos con tres cambios de sección decreciente para nivelar cada etapa.

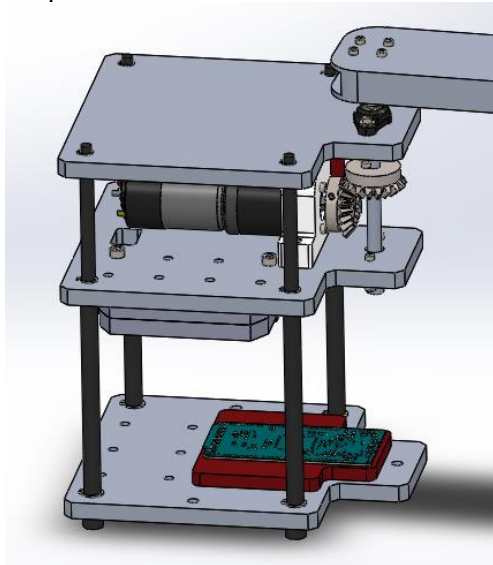


Fig. 20. Estructura base para montaje vertical, soportada en cuatro postes de aluminio

Además, se generó un patrón de perforaciones en las dos etapas inferiores, con el fin de actuar como puntos de fijación rápida de los componentes electrónicos mediante la fabricación de bases plásticas coincidentes con el patrón.

El eje que transmite el movimiento al primer eslabón móvil, también se modificó para estar conformado por dos secciones longitudinales de 8mm de diámetro que se unen entre sí mediante tornillo, de tal manera que su montaje sea vertical al igual que el resto de la estructura.

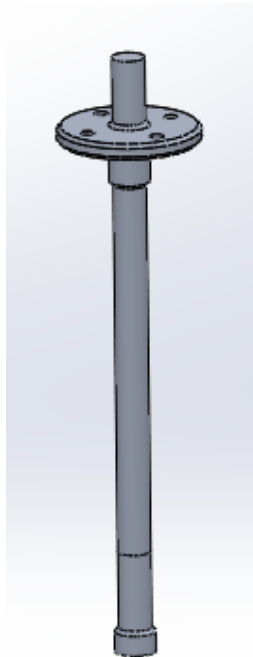


Fig. 21. Eje de transmisión conformado por dos piezas independientes

2. Eslabones

Se adicionaron elementos fabricados por aditiva para brindar soporte a los distintos dispositivos electrónicos necesarios para el control y funcionamiento del robot.

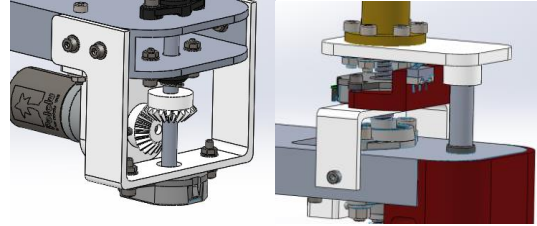


Fig. 22. Piezas adicionales fabricadas por aditiva para soporte de sensores y motores

3. Actuador final

Se realizaron pruebas con un prototipo del primer modelo de la pinza y tras identificar una serie de potenciales complicaciones, se decidió diseñar una distinta, que fuera más sencilla en su construcción y similar en funcionalidad.

Un servomotor permite la apertura o cierre del componente hecho por fabricación aditiva. Al realizar este movimiento, la pieza móvil aprisiona el objeto de interés contra una cara saliente de la base donde se apoya el servo motor.

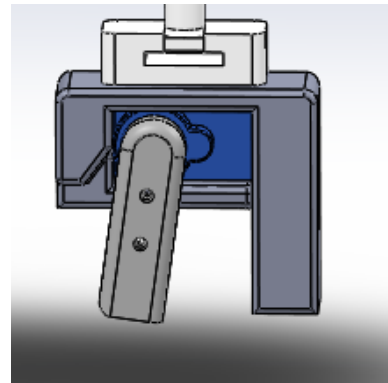


Fig. 23. Modelo definitivo de la pinza

V. CÁLCULOS MECÁNICOS

Tras la optimización del diseño del brazo robótico, se procedió a la realización de los cálculos de los distintos elementos, tanto para comprobar su resistencia como para obtener sus dimensiones y características adecuadas a los esfuerzos a soportar.

A. Análisis estático de esfuerzos

La Fig. 24 muestra, de forma simplificada, los puntos donde se encuentran aplicadas las distintas fuerzas en el robot Scara. Éste se muestra en la situación crítica: cuando los eslabones están completamente alineados y crean mayor momento respecto a la base.

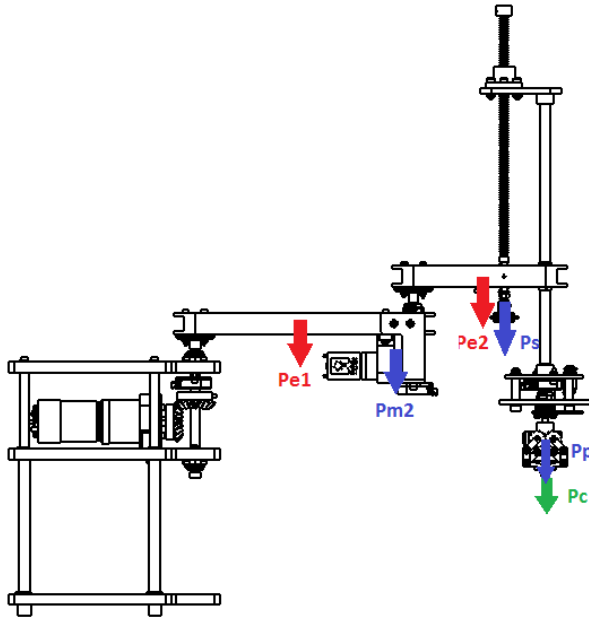


Fig. 24. Esquema simplificado de cargas.

Los valores de las cargas son los siguientes:

- Peso total del robot, $W_r = 60 \text{ N}$
- Carga máxima, $P_c = 10\% \cdot W_r = 6 \text{ N}$
- Peso de la pinza, $P_p = 1 \text{ N}$
- Peso mecanismo de subida, $P_s = 1.5 \text{ N}$
- Peso antebrazo, $P_{e2} = 2.1 \text{ N}$
- Peso motor mediano, $P_{m2} = 1.3 \text{ N}$
- Peso brazo, $P_{e1} = 3.1 \text{ N}$

1. Eslabón 1 (brazo)

Este eslabón podría ser mecanizado a partir de un tubo rectangular de aluminio 6280 de dimensiones exteriores 70x20 mm y 3 mm de espesor en su sección. La dimensión principal es de 195 mm. Está sometido a fuerzas ejercidas por el peso propio y los pesos de otros componentes en el mismo eslabón, así como al momento que crean los otros elementos estructurales del robot Scara y la carga que efectúa el objeto a trasladar.

En la Fig. 25 se puede ver la deformación en la dirección Y (vertical) del eslabón en el análisis estático realizado mediante el método de elementos finitos en SolidWorks. La deformación máxima, como era de esperar ya que está en voladizo, se produce en el extremo derecho del eslabón y tiene un valor de 0.42 mm. En la Fig. 26 se puede ver la distribución de tensiones de Von Mises. El valor máximo es de 68.96 MPa mientras que el límite elástico del material es 260 MPa. Este perfil es adecuado, ya que, aunque se aplique un factor de seguridad de 2, no se sobrepasa el límite elástico del aluminio 6082.

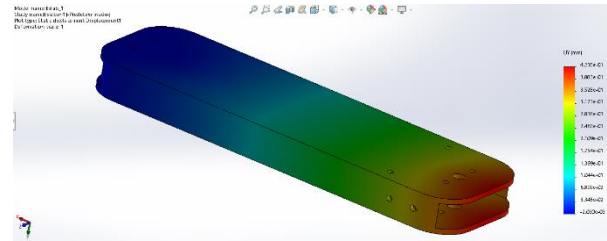


Fig. 25. Deformaciones en el brazo en dirección vertical.

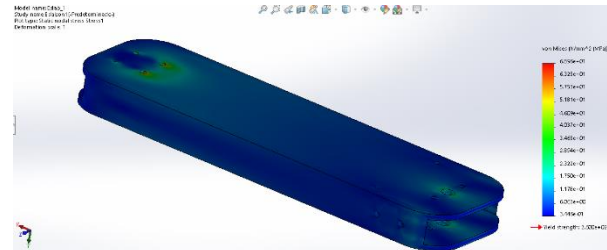


Fig. 26. Distribución de tensiones de von Mises en el eslabón 1.

2. Eslabón 2 (antebrazo)

Para este eslabón se utilizaría el mismo perfil que para el anterior. La dimensión principal de éste es 128 mm. Está sometido a un tipo de fuerzas similar, pero de valor inferior debido a su localización.

En la Fig. 27 se muestra la deformación en la dirección Y (vertical) del eslabón en el análisis estático. Al igual que en el brazo, la deformación máxima se produce en el extremo derecho y tiene un valor de 0.01 mm. La máxima tensión de Von Mises es 5.27 MPa (ver Fig. 28), así que este caso es aún más favorable que el del brazo.

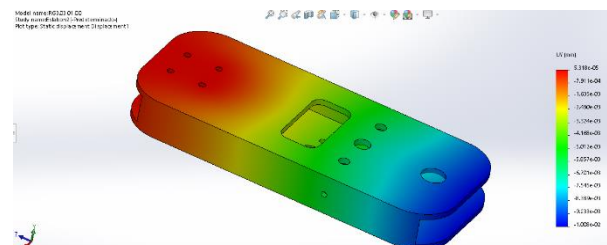


Fig. 27. Deformaciones en el antebrazo en dirección vertical.

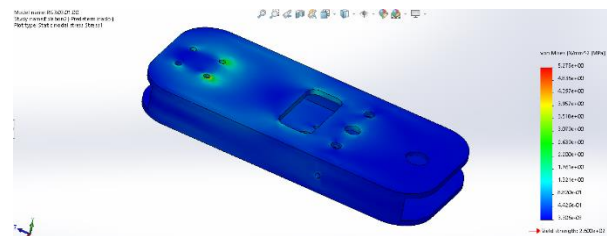


Fig. 28. Distribución de tensiones de von Mises en el antebrazo.

B. Cálculo de engranajes

Cálculo de fuerzas

La fuerza tangencial que soporta el engranaje cuando el motor funciona a su máxima potencia se calcula con (14), para la cual se necesitan los siguientes parámetros geométricos tanto para el engranaje motor como para el conducido:



- número de dientes del engranaje motor y del engranaje conducido: Z_{motor} y $Z_{\text{conducido}}$, respectivamente
- módulo de los engranajes: m
- ángulo de presión: α
- ancho del diente: b
- altura del diente: h
- ángulo de paso del engranaje motor

$$tg(\delta_1) = Z_{\text{motor}}/Z_{\text{conducido}} \quad (1)$$

$$tg(\delta_2) = Z_{\text{conducido}}/Z_{\text{motor}} = \Sigma - \delta_1 \quad (2)$$

- radio primitivo

$$r_p = \frac{m}{2} \cdot Z \quad (3)$$

- radio medio

$$r_m = r_p - \frac{b}{2} \cdot \cos(\delta_i) \quad (4)$$

y el par ejercido por el motor, obtenido tras la caracterización de los diferentes motores.

En engranajes cónicos, las fuerzas axial y radial se calculan a partir de la fuerza tangencial como se indica en (6) y (7)

$$F_t = T/r_m \quad (14)$$

$$F_a = F_t \cdot tg(\alpha) \cdot \sin(\delta) \quad (6)$$

$$F_r = F_t \cdot tg(\alpha) \cdot \cos(\delta) \quad (7)$$

En engranajes rectos, la fuerza normal se calcula a partir de la fuerza tangencial como se indica en (8) y (9)

$$F_t = T/r_m \quad (8)$$

$$F_n = F_t \cdot tg(\alpha) \quad (9)$$

Una vez calculada la carga tangencial que se transmite al diente se puede obtener mediante (10)(2) la tensión aplicada en el diente, σ , siendo Y el factor de forma o de Lewis, que para un ángulo de presión de 20° se calcula (11). Esta tensión σ no debe superar la tensión admisible del material, σ_{adm} .

$$F_t = \sigma \cdot b \cdot m \cdot Y \quad (10)$$

$$Y = \frac{Z}{7 \cdot (Z + 8)} \quad (11)$$

Resistencia de los dientes a flexión

Para el análisis a flexión se considera que solamente hay contacto por parte de un diente de cada engranaje y que el diente es una barra en voladizo en la que se aplica la fuerza tangencial en el extremo del diente. Con estas consideraciones, el resultado dado por (12) es una aproximación de lado de la seguridad.

$$\sigma = M/W \quad (12)$$

El momento flector, M , se calcula mediante (13) y W , que es el módulo resistente de la sección en la base del diente, se calcula mediante (14).

$$M = F_t \cdot h \quad (13)$$

$$W = \frac{I}{y_m} \quad (14)$$

Donde I es el momento de inercia del perfil del diente respecto al eje neutro de la sección, siendo s el espesor de la base del diente y y_{max} es la distancia del eje neutro de la sección transversal del diente a la fibra más alejada (16).

$$I = b \cdot \frac{s^3}{12} \quad (15)$$

$$y_{\text{max}} = \frac{s}{2} \quad (16)$$

1. Articulación A

La transmisión del movimiento del motor E192-24-336, el cual se encuentra en la base del robot, se realiza mediante engranajes cónicos impresos en PLA mediante fabricación aditiva. Las características de éstos vienen dadas en la TABLA IV. .

TABLA IV. CARACTERÍSTICAS ENGRANAJES ARTICULACIÓN A

Número de dientes, Z	18
Módulo, m (mm)	2
Ángulo de presión, α ($^\circ$)	20
Ancho del diente, b (mm)	8
Altura del diente, h (mm)	8.7
Espesor del diente en la base, s (mm)	4.4

TABLA V. TENSIONES A SOPORTAR POR LOS ENGRANAJES DE LA ARTICULACIÓN A

σ (MPa)	33.99
$\sigma_{\text{flexión}}$ (MPa)	18.13

2. Articulación B

La transmisión del movimiento del motor 378:1, el cual se encuentra en el eslabón 2 del robot, se realiza mediante engranajes cónicos impresos en PLA mediante fabricación aditiva. Las características de éstos vienen dadas en la TABLA VI. .

TABLA VI. CARACTERÍSTICAS ENGRANAJES ARTICULACIÓN B

Número de dientes, Z	20
Módulo, m (mm)	2
Ángulo de presión, α ($^\circ$)	20
Ancho del diente, b (mm)	6



Altura del diente, h (mm)	5.7
Espesor del diente en la base, s (mm)	2.26

$\sigma_{\text{flexión}}$ (MPa)	15.17
---------------------------------	-------

La tensión a soportar es menor que la admisible del material en todos los casos.

TABLA VII. TENSIONES A SOPORTAR POR LOS ENGRANAJES DE LA ARTICULACIÓN B

σ (MPa)	16.08
$\sigma_{\text{flexión}}$ (MPa)	21.97

3. Articulación C

La transmisión del movimiento del motor 1000:1, el cual se encuentra en el antebrazo del robot, se realiza mediante engranajes cilíndricos rectos comerciales. Las características de éstos vienen dadas en la TABLA VIII. .

TABLA VIII. CARACTERÍSTICAS ENGRANAJES ARTICULACIÓN C

Número de dientes, Z	50
Módulo, m (mm)	1
Ángulo de presión, α (°)	20
Ancho del diente, b (mm)	2
Altura del diente, h (mm)	1.1
Espesor del diente en la base, s (mm)	1.05

TABLA IX. TENSIONES A SOPORTAR POR LOS ENGRANAJES DE LA ARTICULACIÓN C

σ (MPa)	19.75
$\sigma_{\text{flexión}}$ (MPa)	14.56

4. Articulación D

La transmisión del movimiento del motor 1000:1, el cual se encuentra en la articulación D, se realiza mediante engranajes cilíndricos rectos comerciales. Las características de éstos vienen dadas en la TABLA X. .

TABLA X. CARACTERÍSTICAS ENGRANAJES ARTICULACIÓN D

Número de dientes, Z	48
Módulo, m (mm)	1
Ángulo de presión, α (°)	20
Ancho del diente, b (mm)	2
Altura del diente, h (mm)	1.1
Espesor del diente en la base, s (mm)	1.05

TABLA XI. TENSIONES A SOPORTAR POR LOS ENGRANAJES DE LA ARTICULACIÓN D

σ (MPa)	20.69
----------------	-------

C. Cálculo de árboles

Se ha realizado asimismo el cálculo del diámetro necesario de los ejes para la transmisión del movimiento de los motores a los eslabones. El predimensionado se ha llevado a cabo mediante la expresión simplificada en (19), donde M_t representa la suma cuadrática del momento torsor y flector a los que está sometido el árbol.

$$\tau_t = \frac{M_t}{W_t} \quad (26)$$

$$W_t = \pi \cdot \frac{d^3}{16} \quad (27)$$

$$d \geq \sqrt[3]{\frac{16 \cdot M_t}{\pi \cdot \tau_t}} = 1.72 \cdot \sqrt[3]{\frac{M_t}{\tau_t}} \quad (19)$$

En la TABLA XII. , y a partir de los datos de caracterización de cada uno de los motores y el material empleado para su fabricación, se recogen los valores obtenidos para los diámetros junto a los diámetros normalizados, que serán los que se utilicen en la fabricación de los correspondientes ejes.

TABLA XII. DIÁMETROS MÍNIMOS PARA LOS ÁRBOLES

Pieza	d (mm)	Valor normalizado (mm)
RG3.01.0 4.01/02	7.04	8
RG3.02.0 4.00	5.13	6
Husillo	3.52	4*
RG3.04.0 3.01	5.77	6

(*) Por lo tanto, el husillo de diámetro 8 mm es aceptable.

D. Cojinetes articulados

Los cojinetes articulados empleados son capaces de soportar las fuerzas axiales y radiales que el eje les transmite hasta los valores máximos que se muestran en la TABLA XIII. .

TABLA XIII. CARGAS SOPORTADAS POR LOS COJINETES ARTICULADOS



Referencia comercial del cojinete articulado	Carga axial máxima permitida (N)
EFOM-06	500
EFOM-08	700

Las cargas radiales que deben soportar son debidas a la fuerza que los engranajes producen sobre cada eje, así como a los momentos flectores aplicados debidos al peso de otros componentes. En cambio, las fuerzas axiales serán debidas al peso de cada eje y a cómo se haya realizado el diseño. En la TABLA XIV. se muestran los valores de las cargas radiales para cada uno de los cojinetes articulados. Las cargas axiales son prácticamente despreciables, excepto por el cojinete en la articulación D que es el que más carga axial soporta debido a la pinza y la carga a mover (unos 7 N).

TABLA XIV. CARGAS APLICADAS A LOS COJINETES ARTICULADOS

Articulación	Tipo	Cojinete	Fuerza radial (N)
A	EFOM-08	Inferior	22.27
		Superior	76.66
B	EFOM-06	Inferior	10.66
		Superior	30.57
D	EFOM-08	-	6.47

Los esfuerzos a los que está sometido cada cojinete en la peor situación no supera en ningún momento la carga máxima permitida. Lo mismo ocurre en el caso de las cargas axiales. Por tanto, estos cojinetes articulados cumplen su función adecuadamente.

No se dispone de datos sobre los cojinetes empleados en la articulación C, pero las fuerzas a las que están sometidos son también bajas.

VI. SELECCIÓN DE ELEMENTOS Y COMPONENTES ELECTRÓNICOS

Como ya se ha mencionado, de acuerdo a los cálculos obtenidos del modelado y simulación del robot, se han seleccionado motores DC para las articulaciones A, B, C y el movimiento lineal. Ya que para la pinza no se requiere ningún tipo de control, se ha trabajado con un servomotor y así simplificar su apertura/cierre.

Para realizar el control de los motores DC se requiere contar con una retroalimentación de su posición por tal razón se ha decidido utilizar encoders absolutos de la marca CUI referencia AMT 203-1024-8000W (2 unidades) y 203-1024-6000W (2 unidades).

Adicionalmente, se requiere utilizar drivers para el control de los motores DC ya que el microcontrolador a utilizar no es capaz de proporcionar la corriente necesaria para mover los motores.

Para realizar el control del robot se utilizó la plataforma de desarrollo Arduino debido a la facilidad de configuración e implementación en el desarrollo de este tipo de proyectos. Se ha decidido trabajar con un Arduino DUE, que está basado en un microcontrolador Atmel SAM3X8E ARM Cortex-M3 de 32-bit de alto rendimiento, que opera a una velocidad máxima de 84 MHz con una memoria Flash de 512 Kbytes, esto debido a las altas exigencias de procesamiento de datos que supone el control de los 4 motores DC, el servomotor y la retroalimentación a través de los encoders, además de la comunicación de estos encoders y la interfaz gráfica.

Como anexos, también se entregan los datasheet de los motores, encoders y microcontrolador del Arduino DUE.

VII. DISEÑO DE LA TARJETA ELECTRÓNICA DE CONTROL

Como primera medida se ha realizado un listado de señales necesarias para el control del robot SCARA, que se entrega como archivo adjunto, con el cual es posible identificar cuántas, cuáles y de qué tipo de señales son necesarias. También se ha registrado en él, qué tipo de conector y/o componente es requerido en la placa, esto es de gran utilidad para mantener el orden cuando se vaya a realizar el ensamblado, ya que se han utilizado componentes electrónicos que se disponían en el laboratorio de electrónica evitando tener que comprar componentes adicionales. Dicho listado también ha permitido determinar si los puertos de I/O del Arduino son suficientes para realizar las labores correspondientes.

A. Entradas/Salidas digitales

Se han incluido 6 microswitch final de carrera para delimitar los movimientos máximos y mínimos de cada eslabón del robot. Además, las 4 señales de los Chip Select de cada encoder, ya que como se explicará más adelante, las lecturas de estas señales se realizarán vía comunicación SPI. Dos selectores de palanca se han incluido, uno para el cierre/apertura de la pinza y otro para la configuración del modo en el que se desea operar (Manual → moviendo x grados cada eslabón. Automático → mediante coordenadas x, y, z). Una señal para conectar un paro de emergencia para el robot, 4 leds para indicar algunas configuraciones deseadas y 1 conector adicional dejado como reserva en caso de ser necesaria alguna señal que no se haya tenido en cuenta. La TABLA XV. muestra un resumen de dichas entradas.

TABLA XV. LISTADO DE ENTRADAS/SALIDAS DIGITALES

Señal	Pin Arduino	Tipo de señal
Final 1 Eslabón 1 - Hombro	39	DI
Final 2 Eslabón 1 - Hombro	41	DI
Final 1 Eslabón 2 - Codo	43	DI
Final 2 Eslabón 2- Codo	45	DI
Final 1 Eslabón 3 - Subida	47	DI



Final 2 Eslabón 3 - Subida	49	DI
Encoder 1 - CS	36	DI
Encoder 2 - CS	34	DI
Encoder 3 - CS	32	DI
Encoder 4 - CS	30	DI
Paro de emergencia	22	DI
Apertura-Cierre pinza	24	DI
Selección de modo manual- auto	38	DI
Adicional	40	DI
LED 1	42	DO
LED 2	44	DO
LED 3	46	DO
LED 4	48	DO

B. Entradas/Salidas analógicas

Se ha decidido conectar 4 potenciómetros para el movimiento de cada articulación cuando se tiene seleccionado el control manual del robot.

TABLA XVI. LISTADO DE ENTRADAS/SALIDAS ANALÓGICAS

Señal	Pin Arduino	Tipo de señal
Potenciómetro 1 - hombro	A0	AI
Potenciómetro 2 - codo	A1	AI
Potenciómetro 3 - subida	A2	AI
Potenciómetro 4 - muñeca	A3	AI

C. PWM

El control de cada motor de corriente continua se realizará mediante la modulación de ancho de pulsos PWM, que será generada por el Arduino pero que luego será enviada a un driver TB9051FTG, como el mostrado en la Fig. 29 que básicamente contiene 1 puente H completo con el cual es posible realizar el control de los motores CC. De esta manera se tienen 4 driver para el gobierno de los 4 motores CC y el servomotor para la pinza será controlado directamente desde uno de los pines del Arduino, como se muestra en la TABLA XVII.



Fig. 29. Driver TB9051FTG para motor de escobillas DC

TABLA XVII. LISTADO DE SALIDAS DE PWM

Señal	Pin Arduino	Tipo de señal
PWM1 Motor 1	10	PWM
PWM2 Motor 1	11	PWM
PWM1 Motor 2	8	PWM
PWM2 Motor 2	9	PWM
PWM1 Motor 3	6	PWM
PWM2 Motor 3	7	PWM
PWM1 Motor 4	4	PWM
PWM2 Motor 4	5	PWM
PWM1 Motor 5	12	PWM

La Fig. 30 muestra las señales necesarias para utilizar el driver mencionado de acuerdo al control que se quiera realizar al motor.

TB9051FTG simplified truth table (PWM1 + PWM2)						
Inputs				Outputs		Operation
EN	ENB	PWM1	PWM2	OUT1	OUT2	
1	0	PWM	0	PWM (H/L)	L	forward/brake at speed PWM %
		0	PWM	L	PWM (H/L)	reverse/brake at speed PWM %
		0	0	L	L	brake low (outputs shorted to ground)
		1	1	L	L	brake low (outputs shorted to ground)
0	X	X	X	Z	Z	coast (outputs floating/disconnected)
X	1	X	X	Z	Z	

Fig. 30. Tabla de verdad para la configuración de la salida del Driver TB9051FTG

D. Comunicación

Se ha planteado utilizar un LCD en el cual se pretenden mostrar los parámetros de funcionamiento más importantes del robot, pero debido a la cantidad de señales que van al Arduino se ha decidido utilizar un módulo I2C para conectarlo a través de él y liberar pines del Arduino, por tal razón se ha dejado un conector para la comunicación I2C entre el Arduino y el LCD. También se han dejado en la placa dos conectores de comunicación serial en caso de ser necesaria la comunicación con otro Arduino y/u otro componente de hardware adicional. Además, debido a que se están utilizando encoders absolutos, estos son capaces de brindar su posición absoluta, pero dicha lectura debe realizarse vía comunicación SPI (Serial Peripheral Interface), que es una comunicación serial síncrona full duplex, idónea para aplicaciones en la que se requiere transferencia de datos entre circuitos electrónicos que se encuentran a cortas distancias. Por tal razón se ha utilizado el puerto previsto en el Arduino para dicha comunicación. La TABLA XVIII. contiene las señales de comunicación asociadas al circuito diseñado.

TABLA XVIII. LISTADO DE ENTRADAS/SALIDAS DE COMUNICACIÓN

Señal	Pin Arduino	Tipo de señal
Comunicación serial 1 Rx	14	COM SER



Comunicación serial 1 Tx	15	COM SER
Comunicación serial 2 Rx	16	COM SER
Comunicación serial 2 Tx	17	COM SER
Comunicación SPI VCC	ICSP VCC	COM
Comunicación SPI GND	ICSP GND	COM
Comunicación SPI MISO	ICSP 1	COM
Comunicación SPI MOSI	ICSP 4	COM
Comunicación SPI SCK	ICSP 3	COM
Comunicación SPI RESET	ICSP RESET	COM
I2C LCD	20	COM I2C
I2C LCD	21	COM I2C

Una vez definida claramente de manera cualitativa y cuantitativamente las señales necesarias para el gobierno del robot, se procedió al diseño del shield para el Arduino. Al momento de diseñar la placa se ha tenido en cuenta colocar resistencias de pull-up de 4.7 k Ω y resistencias de limitación de corriente de 330 Ω para las entradas y salidas digitales, respectivamente.

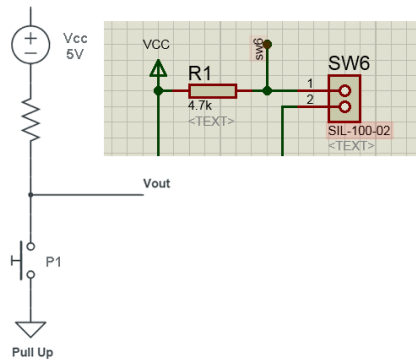


Fig. 31. Resistencia de Pull Up

Adicionalmente se han colocado dos condensadores de 100 nF (uno electrolítico y uno cerámico) de desacoplo de altas frecuencias para eliminar el ruido a la entrada de alimentación DC.

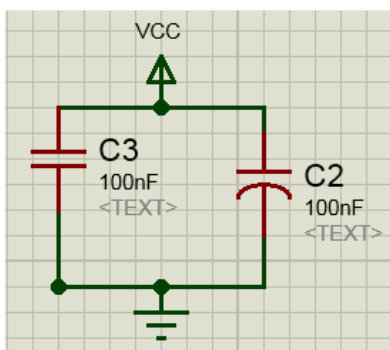
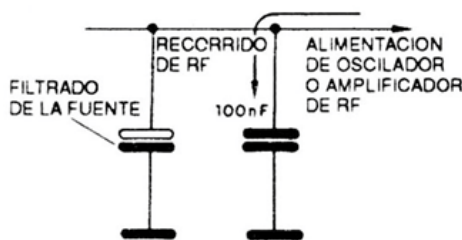


Fig. 32. Circuito de desacoplo de la señal de entrada de 5 VCC

Para el diseño de dicha tarjeta electrónica de control se ha utilizado el software Proteus, en él se ha realizado el diseño del circuito necesario, así como el diseño de la PCB, como se muestra en las Fig. 33, Fig. 34, Fig. 35 y Fig. 36

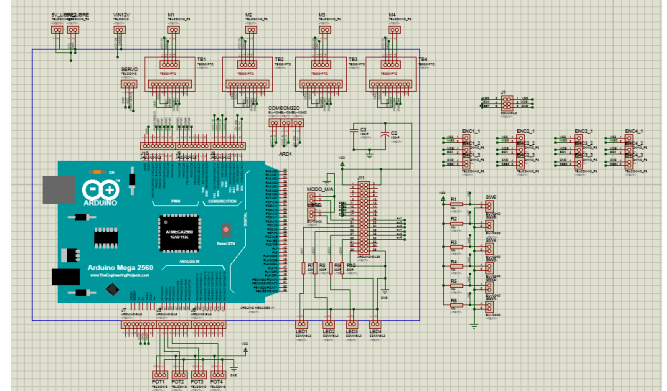


Fig. 33. Diseño de circuito de control para el robot con Arduino

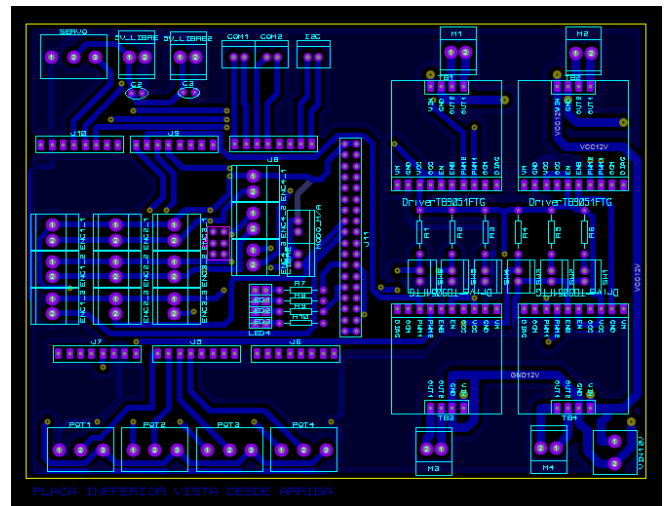


Fig. 34. Diseño de PCB capa inferior

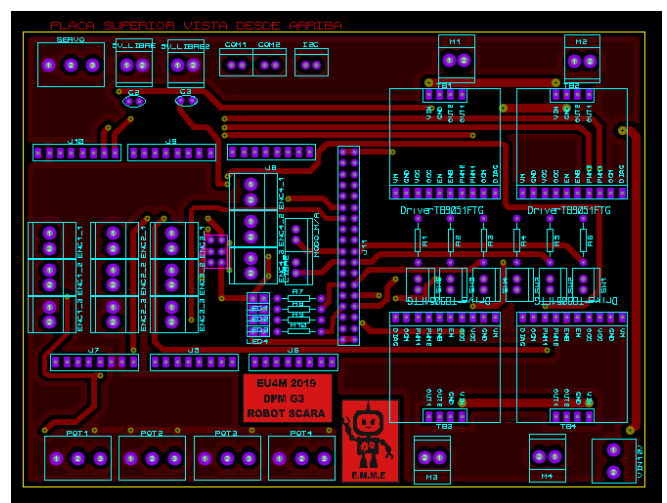


Fig. 35. Diseño de PCB capa superior

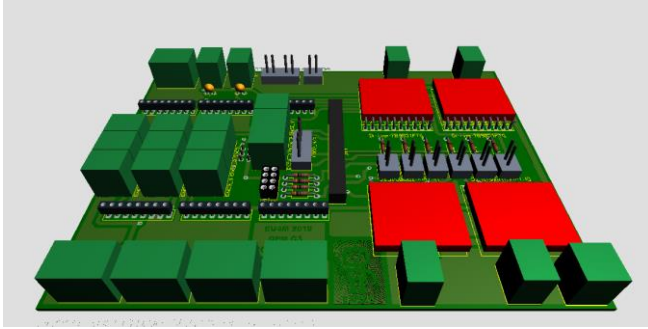


Fig. 36. Vista esquemática 3D del Shield diseñado

Una vez obtenida la PCB para el Shield del Arduino se envió a imprimir y fabricar. Sin embargo, en esta primera fabricación se tuvo un problema y es que los componentes no se ajustaban al tamaño real de la placa. Posteriormente se evidenció que el problema radicó en un ajuste de escala en el momento de la impresión del esquemático. Se procedió a realizar la corrección y nueva impresión. Dicha placa tenía las dimensiones necesarias, pero no fue posible utilizarla, ya que en dicho momento un no se estaba contemplando la lectura de las posiciones absolutas del encoder vía SPI, por lo que fue necesario realizar una versión 3 del circuito electrónico y su correspondiente PCB, dicho circuito es el que se está utilizando actualmente en el robot y es el que refleja las señales y pines ya descritos anteriormente.

La Fig. 37 muestra el circuito final construido.

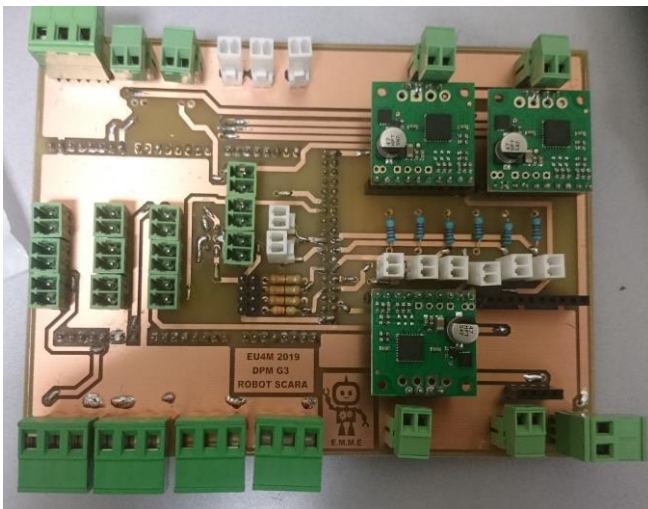


Fig. 37. Circuito de control final

VIII. ENSAMBLAJE Y MONTAJE FÍSICO DEL ROBOT

El montaje del robot se realizó por subensambles, ya que estaba diseñado de esta forma para que resultara más sencillo. Se comenzó desde el último eslabón (la pinza) hacia la base del robot.

Las piezas fabricadas por aditiva de la pinza fueron de las primeras en estar disponibles, así que una vez que se recibió el servomotor pudo realizarse el montaje de este subensamblado, que se muestra en la Fig. 38

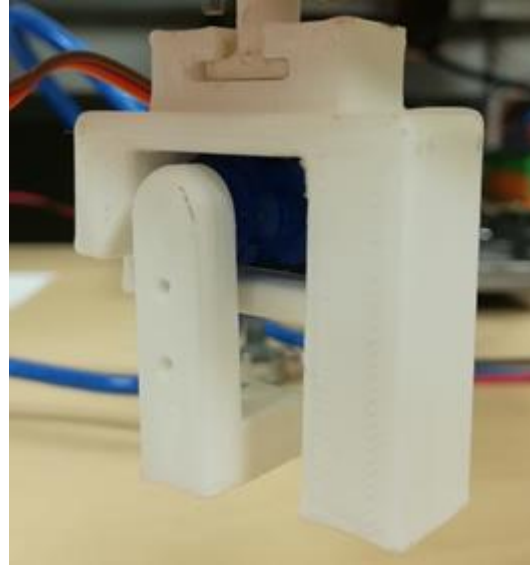


Fig. 38. Montaje de la Pinza

Posteriormente, y tras haber recibido los pedidos comerciales tanto de los componentes mecánicos como de los electrónicos, se pudo proceder al ensamblaje del movimiento prismático en partes. La primera parte de este montaje consistió en atornillar la tuerca del husillo a la pieza correspondiente para unirlo a su vez con el tubo y en insertar ésta en el husillo. La segunda parte consistió en colocar el encoder, el motor, el cojinete articulado y los engranajes en las piezas impresas mediante tornillos y añadiendo unos separadores. En la parte inferior de la Fig. 39 se ve la pieza mediante la cual este subensamblado se une a la pinza por encaje.

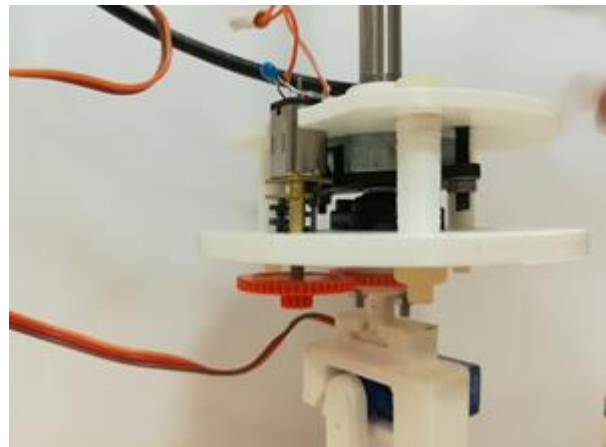


Fig. 39. Montaje del sistema de rotación (muñeca)

El subensamblado del antebrazo fue el siguiente. Como en el anterior, se colocaron los cojinetes, el motor y el engranaje en el eje de salida. Posteriormente se colocó el encoder en su soporte y se atornilló éste al eslabón.

En ese momento pudo efectuarse la unión entre el mecanismo del movimiento prismático y del antebrazo (Fig. 40): se insertó el husillo a través del encoder y de los cojinetes y a continuación se añadió el engranaje al extremo libre del husillo. Al mismo tiempo, se insertó el tubo a través del correspondiente hueco del eslabón para unir en su extremo



inferior el mecanismo encargado de realizar el giro de la pinza.

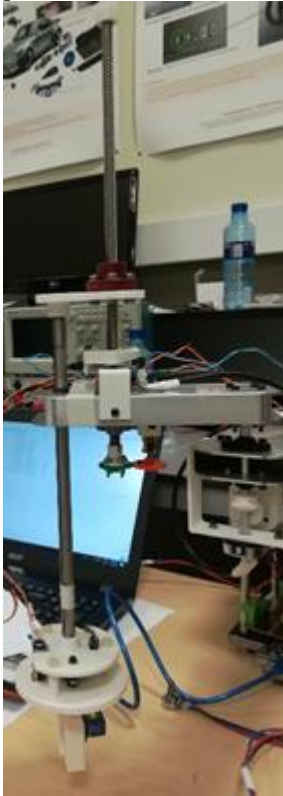


Fig. 40. Montaje del mecanismo de subida

El montaje del antebrazo comenzó por la colocación de los cojinetes articulados a través de los cuales se introdujo el eje y al cual se fijó el engranaje correspondiente mediante un prisionero. Por separado, el motor se dispuso en la abrazadera con su engranaje y el encoder se atornilló a su soporte. Para completar el subensamblé, se unieron ambos al eslabón y este se unió mediante el eje al resto de piezas ya ensambladas (Fig. 41).



Fig. 41. Montaje del antebrazo

Por último, la base (Fig. 42). Como ya se describió anteriormente, la base permite un montaje vertical, ya que las placas se insertan en los postes desde la parte superior para formar una estructura. Se colocó el motor en su abrazadera y, junto con el engranaje ya en el eje del motor, se atornilló ésta en la repisa central. Tras colocar los cojinetes articulados, se

insertó a través de éstos y del engranaje el eje. Finalmente se atornilló el eje al brazo para completar el robot. La Fig. 43 muestra una vista desde la parte superior.

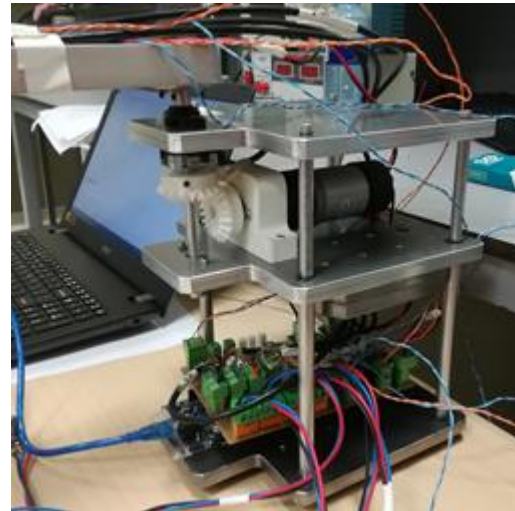


Fig. 42. Montaje del brazo en la base del robot



Fig. 43. Vista superior del robot

Adicionalmente se colocaron algunas piezas para mejorar las características del robot (soportes para los finales de carrera o los potenciómetros, una base para el Arduino...).

Una vez montados todos los elementos mecánicos se instaló la placa de control electrónica en la base del robot, dejándolo todo embebido como se había planteado desde el inicio.

IX. CINEMÁTICA INVERSA DEL ROBOT

Nuestro Robot tiene la siguiente distribución de sus eslabones de manera esquemática, con sus sistemas de referencia dispuestos de esa forma.

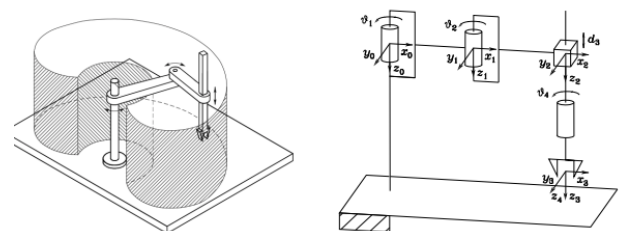


Fig. 44. Robot Scara y sus sistemas de referencia.



En función de la disposición del robot de la Fig. 44 se expresa a continuación la tabla de los parámetros de Denavit Hartenberg de nuestro Robot Scara.

TABLA XIX. DATOS EXPERIMENTALES MOTOR 1

n	α_i	a_i	d_i	Theta i
1	0	a_1	d_1	q_1
2	π	a_2	d_2	q_2
3	0	0	q_3	0
4	0	0	d_4	q_4

De donde q_1, q_2, q_3, q_4 son los ángulos que forman entre los eslabones, y d_1, d_2 , son las distancias en Z que hay entre eslabones; a_1, a_2 son las longitudes de los eslabones definidas previamente. En función de estos parámetros se obtiene la matriz de transformación. Sin embargo, no se detalla sus cálculos ya que por facilidad se propone el cálculo cinemático inverso mediante relaciones geométricas en base a la

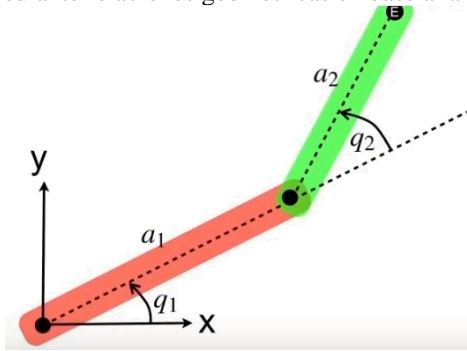


Fig. 45. Robot Scara y sus sistemas de referencia.

Sin embargo, debido a facilidad de implementación se ha desarrollado la cinemática inversa del robot mediante métodos geométricos [1], las ecuaciones que permiten obtener los ángulos se describen a continuación:

$$q_2 = \cos^{-1} \left(\frac{x^2 + y^2 - a_1^2 - a_2^2}{2 * a_1 * a_2} \right) \quad (29)$$

$$q_1 = \tan^{-1} \left(\frac{y}{x} \right) - \tan^{-1} \left(\frac{a_2 * \sin(q_2)}{a_1 + a_2 * \cos(q_2)} \right) \quad (30)$$

El valor de q_3 que en este caso es un movimiento prismático vendría dado por:

$$q_3 = h - z \quad (31)$$

Y finalmente la orientación de la pinza, que actúa como TCP, sería la siguiente:

$$q_4 = \psi - \tan^{-1} \left(\frac{y}{x} \right) \quad (32)$$

Donde,

$$\psi = q_1 + q_2 \quad (33)$$

X. MODELADO DE MOTORES

En función de los datos característicos que se han obtenido de los motores previamente para el robot, se procede a determinar cuál sería la manera más adecuada de controlarlos.

Un primer paso en esta sección es realizar un adecuado modelado de los motores para poder tener un modelo que sea muy similar o casi igual al modelo de los motores reales.

Se muestra a continuación el modelado de un Motor DC para poder implementarlo en Matlab y hacer sus respectivas pruebas.

Considerando el circuito del Motor descrito por

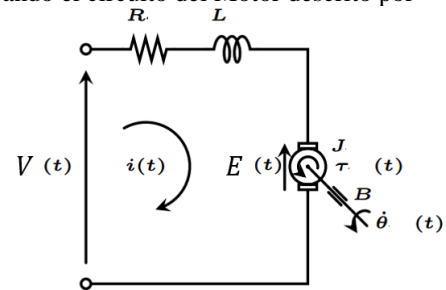


Fig. 46. Modelo de Motor DC.

La ecuación eléctrica del motor tiene la forma

$$V(t) = R * i(t) + L * \frac{di(t)}{dt} + E \quad (34)$$

De donde, $V(t)$ es la tensión de entrada del motor, $i(t)$ la corriente, E la fuerza contraelectromotriz, R la resistencia nominal y L la inductancia del rotor.

Por otro lado, la ecuación mecánica del motor es:

$$T(t) = J * \dot{\omega}(t) + B * \omega(t) + T_f \quad (35)$$

De donde J la inercia del rotor, $\omega(t) = \dot{\theta}(t)$ es la velocidad angular del motor, $B\omega(t)$ el par de fricción viscosa con B la constante de fricción viscosa, y T_f el par que incorpora la carga y la fricción.

Se considera un motor DC que satisface las siguientes relaciones:

$$E = k * \omega(t) \quad (36)$$

$$T = k * i(t) \quad (37)$$

Una vez definido lo anterior se obtiene la función de transferencia del Motor sin carga. Reemplazando (36) y (37) en las ecuaciones eléctrica y mecánica (34) y (35).

$$V(t) = R * i(t) + L * \frac{di(t)}{dt} + k * \omega(t) \quad (38)$$



$$T = J * \dot{\omega}(t) + B * \omega(t) \quad (39)$$

Aplicando la trasformada de Laplace

$$V(s) = (R + Ls) * i(s) + k * \omega \quad (40)$$

$$K * i(s) = (Js + B) * \omega(s) \quad (41)$$

Con la transformada de Laplace a las ecuaciones del motor bajo condiciones iniciales nulas, y eliminando la variable de la corriente eléctrica $i(s)$ se obtiene la función de transferencia del motor

$$G(s) = \frac{\omega(s)}{V(s)} = \frac{k}{(Js + B) * (R + Ls) + k^2} \quad (42)$$

En función de esto el motor propuesto en Simulink es el que se muestra en Fig. 47.

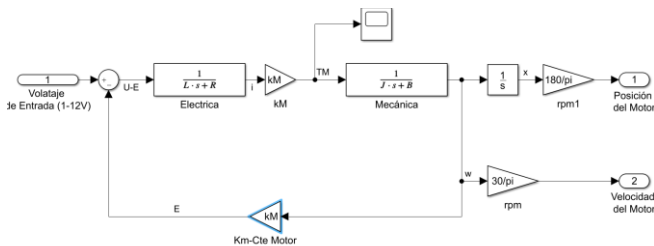


Fig. 47. Modelo Motor DC en Simulink.

Entonces se calculan a continuación los parámetros necesarios de manera experimental, ya que el fabricante de los motores no nos da todas sus características para poder establecer este modelo de manera experimental tomando datos reales del motor, que se muestran en las siguientes tablas.

A. Motor del Brazo (E192.24.36)



Fig. 48. Motor del Codo

Se presenta a continuación una tabla de datos tomado del motor, ya que el fabricante de los motores no nos da todas sus características para poder establecer los parámetros característicos para su modelado, entonces se lo hará de manera forma experimental tomando datos reales del motor,

El motor mencionado en secciones previas es el moto-reductor E192.24.36, el cual tiene como tensión máxima de 24 V, un torque máximo de 3 [Nm], una velocidad a 24 V de 11.5 rpm y una corriente máxima de 0.34 [A], sin embargo, su alimentación será de 12 [V], ya que se ha considerado que el robot tendrá una única fuente de este valor para la parte de potencia, los datos esenciales obtenidos del motor se muestran en la TABLA XX.

TABLA XX. DATOS EXPERIMENTALES MOTOR 1

V [v]	I [A]	f [pul-sos/s]	W [rpm]	W [rad/s]	E [v]
1.3	0.09	0	0	0	0.07
2	0.09	11.9	0.71	0.07	0.77
4	0.09	23.81	1.43	0.15	2.77
6	0.1	37	2.22	0.23	4.64
8	0.11	45.45	2.73	0.29	6.50
10	0.11	66.67	4.00	0.42	8.50
12	0.11	73.53	4.41	0.46	10.50

Con estos datos y sabiendo que

$$V = R * i + E \quad (43)$$

La resistencia es

$$R = \frac{V - E}{i} \quad (44)$$

En este motor $R = 13.63 \Omega$, la cual se comprobó midiendo además directamente con un Polímetro en los terminales del Motor.

Luego, se obtiene el valor de la inductancia L , el cuál se lo calculó mediante el método del Oscilador de frecuencias, su valor se determinó alrededor de los 2.83 mH.

Ahora, en cuanto a la parte mecánica del motor, es necesario conocer la constante del motor, el coeficiente de fricción viscoso, y la inercia del motor.

La constante mecánica se la obtiene directamente de la ecuación (36), y reemplazando los valores obtenidos.

$$k = \frac{E}{\omega} = \frac{10.5}{4.41} = 22.82 [V/rpm] \quad (45)$$

La inercia del motor se obtiene en función de la constante de tiempo mecánica (t_m) del motor, mediante la siguiente relación

$$J = \frac{t_m * k^2}{R} \quad (46)$$

Entonces, para conocer la constante de tiempo mecánica se ha utilizado el método del Osciloscopio [2], que consiste en medir el tiempo que tarda en responder el motor a una respuesta paso, en este caso es enviarle los 12 V directamente al motor, como en la Fig. 49

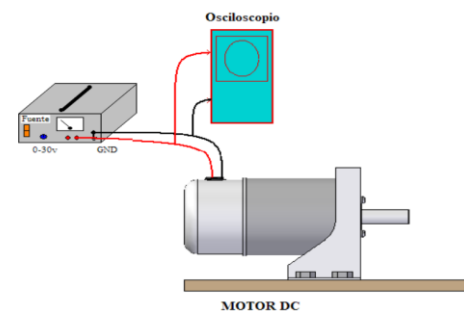




Fig. 49. Método para obtener la constante de tiempo eléctrica.

Después de conectar se debe capturar la imagen del Osciloscopio (Fig. 50) y se ubica el valor del voltaje al 63.2 % para obtener el tiempo que tarda la respuesta en alcanzar el valor deseado.

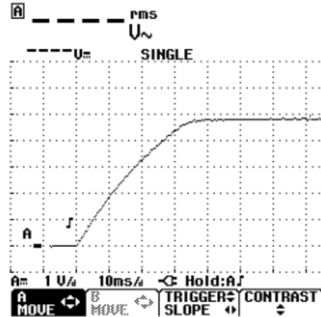


Fig. 50. Determinar constante de tiempo eléctrica en Osciloscopio.

Para el primer motor entonces la constante de tiempo mecánica es 32 ms, y reemplazando en la ecuación (46)

$$J = \frac{0.032 * 22.82^2}{13.63} = 1.22 [kg - m^2] \quad (47)$$

Y finalmente para el cálculo del coeficiente de fricción viscosa se considera la ecuación (48) para estado estable en donde $\dot{\omega} = 0$, que viene de una equivalencia de potencia.

$$E * i = \omega(T_f + T_m) \quad (48)$$

De lo cual T_f es el torque de fricción y $T_m = Ki = 2.51[Nm]$ el torque del motor, se tomarán los datos de la prueba a 12 V, entonces.

$$10.5 * 0.11 = 0.46 * (T_f + 2.51) \quad (49)$$

Y se obtiene que $T_f = 0.35 [Nm]$ y considerando los datos de la TABLA XX. se determina B aproximadamente con la ecuación (51).

$$T_m = k * i = B * \omega + T_f \quad (50)$$

$$2.51 = B * 0.46 + 0.35 \quad (51)$$

Se obtiene que $B = 4.69 [Nm. s]$

Finalmente, las constantes principales del Motor del brazo se resumen en la siguiente tabla.

TABLA XXI. ESPECIFICACIONES DE CONSTANTES

Parámetro	Valor
K	$23 [V/(rad/s)]$
R	13.63Ω
L	$2.83 mH$
t_m	$32 [ms]$
J	$1.22 [kg - m^2]$
T_m	$2.51 [Nm]$

T_f	$0.35 [Nm]$
B	$4.69 [Nm. s]$

De igual manera ingresando este modelo en Matlab con la función de transferencia descrita en (42) y el modelo de la Fig. 47. con una señal de entrada como en la TABLA XX.

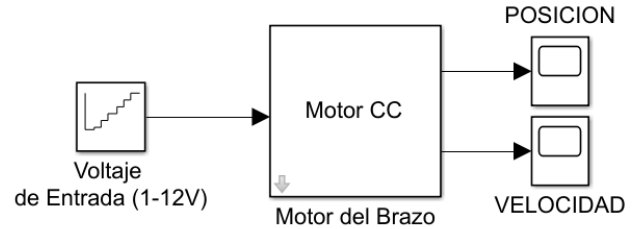


Fig. 51. Motor1 DC con entrada de tensión variable

Se obtienen los siguientes resultados de velocidad Fig. 52, para valores de tensión de entrada entre 1 y 12 V

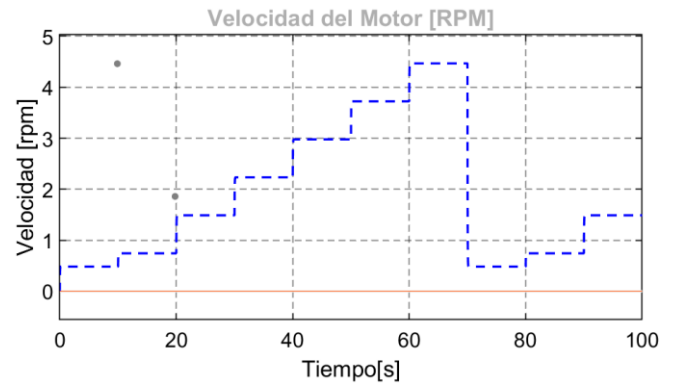


Fig. 52. Velocidad del Motor1 - Modelado

Para ver mejor en relación con los datos experimentales y comprobar que los resultados reales del motor y los de simulación son parecidos, se presenta una comparativa a continuación.

TABLA XXII. DATOS REALES Y DE SIMULACIÓN MOTOR 1

V [v]	W [rpm]	W [rpm]
1.3	0	0.47
2	0.71	0.74
4	1.43	1.49
6	2.22	2.23
8	2.73	2.98
10	4.00	3.73
12	4.41	4.46

Se Comprueba entonces que el modelado es totalmente acorde al motor real, con lo cual se puede proceder a su control, además se verificó no superar el torque máximo del motor de 3 Nm, obteniendo un valor en vacío de 0.73 Nm. Previo al control, se mostrará de manera resumida los datos de los demás motores del robot, los cuales tiene el mismo procedimiento de cálculo.



Considerando que este proceso se realizó para los cuatro motores se presenta a continuación las tablas experimentales de cada uno de los motores.

B. Motor del Antebrazo (378:1 Metal Gearmotor 25Dx58L 12V)



Fig. 53. Motor del Antebrazo

El motor mencionado en secciones previas es el moto-reductor 378:1 Metal Gearmotor 25Dx58L 12V de Pololu, el cual tiene como tensión máxima de 12 V, un torque máximo de 160 [oz-in]/1.13 [Nm], una velocidad de 14 rpm sin carga y una corriente máxima de 1,1 [A], los datos esenciales obtenidos del motor se muestran en la TABLA XXIII.

TABLA XXIII. DATOS EXPERIMENTALES MOTOR 2

V[v]	I[A]	F[pul-sos/s]	W [rpm]	W [rad/s]	E [V]
2	0.07	288.86	2.48	0.26	1.51
4	0.08	567.84	4.87	0.51	3.44
6	0.09	828.10	7.10	0.74	5.37
8	0.09	1092.65	9.37	0.98	7.37
10	0.09	1368.90	11.73	1.23	9.37
12	0.10	1605.50	13.76	1.44	11.30

Y los parámetros obtenidos que siguen el mismo proceso descrito para el motor del Brazo son:

TABLA XXIV. ESPECIFICACIONES DE CONSTANTES MOTOR 2

Parámetro	Valor
K	7.84 [V/(rad/s)]
R	7.1 Ω
L	1.55 mH
t_m	14 [ms]
J	0.123 [kg - m ²]
T_m	0.78 [Nm]
T_f	0.053 [Nm]
B	0.5 [Nm.s]

Ingresando este modelo en Matlab, se obtienen los siguientes resultados de velocidad, para valores de tensión de entrada entre 1 y 12 V

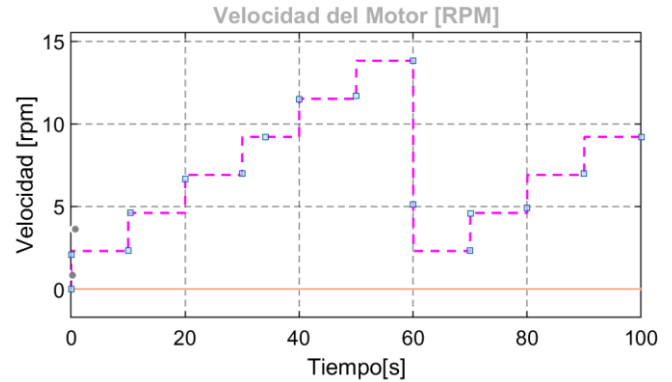


Fig. 54. Velocidad del Motor2 – Modelado

Se presenta una comparativa a continuación entre la simulación y los datos reales.

TABLA XXV. DATOS REALES Y DE SIMULACIÓN MOTOR 2

V [v]	W [rpm] Real	W [rpm] Simulación
2	2.48	2.28
4	4.87	4.6
6	7.10	6.93
8	9.37	9.23
10	11.73	11.55
12	13.76	13.83

Se verificó no superar el torque máximo del motor de 1.13 Nm, obteniendo un valor en vacío de 0.48 Nm.

C. Motor de la Altura y giro de la Pinza (1000:1 - HPCB 12V)



Fig. 55. Motor de control de Altura y giro de la pinza

El motor mencionado en secciones previas es el moto-reductor 1000:1 Micro Metal Gearmotor HPCB 12V de Pololu, el cual tiene como tensión máxima de 12 V, un torque máximo de 140 [oz-in]/0.99 [Nm], una velocidad de 35 rpm sin carga y una corriente máxima de 0.75 [A], los datos esenciales obtenidos del motor se muestran en la TABLA XXVI. TABLA XXIII.

TABLA XXVI. DATOS EXPERIMENTALES MOTOR 3

V[v]	I[A]	F[pul-sos/s]	W [rpm]	W [rad/s]	E [V]
2	0.05	210.00	4.26	0.45	1.32
4	0.06	505.00	10.24	1.07	3.18
6	0.07	781.00	15.84	1.66	5.05
8	0.10	1137.00	23.06	2.42	6.64
10	0.11	1400.00	28.40	2.97	8.50
12	0.11	1678.00	34.04	3.56	10.50



Y los parámetros obtenidos que siguen el mismo proceso descrito para el motor del Brazo y Antebrazo son:

TABLA XXVII. ESPECIFICACIONES DE CONSTANTES MOTOR 3

Parámetro	Valor
K	$2.94 [V/(rad/s)]$
R	5.3Ω
L	$3 mH$
t_m	$9 [ms]$
J	$0.042[kg - m^2]$
T_m	$0.323 [Nm]$
T_f	$0.047 [Nm]$
B	$0.24 [Nm.s]$

De igual manera ingresando el modelo en Matlab, se obtienen los siguientes resultados de velocidad, para valores de tensión de entrada entre 1 y 12 V

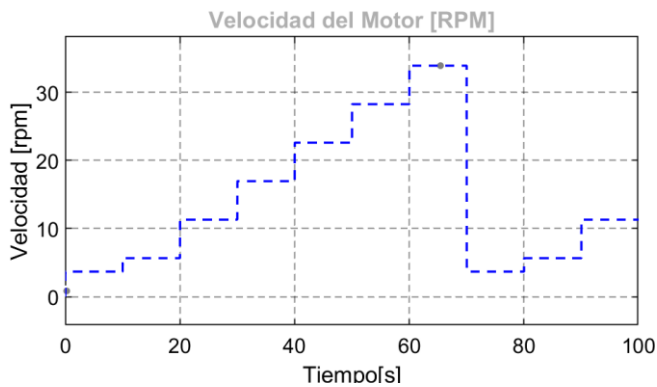


Fig. 56. Velocidad del Motor3- Modelado

Se presenta una comparativa a continuación entre la simulación y los datos reales.

TABLA XXVIII. DATOS REALES Y DE SIMULACIÓN MOTOR 3

V [v]	W [rpm] Real	W [rpm] Simulación
2	4.26	5.67
4	10.24	11.34
6	15.84	16.94
8	23.06	22.61
10	28.40	28.31
12	34.04	33.89

Se verificó no superar el torque máximo del motor de 1 Nm, obteniendo un valor en vacío de 0.28 Nm.

Una vez realizado el modelo de los motores se procede a calibrar su control. Es importante en esta parte aclarar lo siguiente: Inicialmente en el proyecto se consideró hacer una programación de Arduino en tiempo real con Simulink, y se hizo para el control de cada motor de manera individual, sin embargo, debido a que consideramos más sencillo realizar la programación directamente desde el IDE de Arduino, ya que nosotros como programadores tenemos el acceso a casi todas las líneas de código, mientras que en Matlab al tener bloques

y generar un código automático que facilita en ciertas ocasiones programas sencillos, pero en cuanto son programas más largos se complica su configuración para el manejo de un mayor número de motores y variables. Entonces, se explicará brevemente ambos tipos de implementación que se realizaron, después de haber obtenido las constantes de los controladores tipo PID para los 4 motores.

XI. CALIBRACIÓN DE CONTROLADORES EN MATLAB

Se utilizó Matlab para calibrar los controladores, considerando que los modelos de los motores obtenidos previamente son casi iguales a los reales con una variación mínima en su velocidad de salida como ya se demostró antes. Entonces se detalla a continuación el tipo de Control PID implementado.

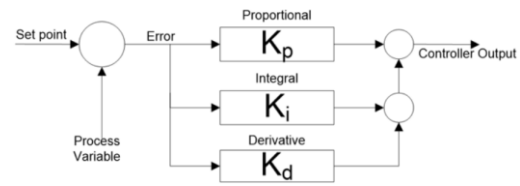


Fig. 57. Controlador Tipo PID

La función de transferencia de un controlador PID estándar es generalmente escrita de la siguiente manera:

$$G(s) = K_p + K_i * \frac{1}{s} + K_D * s \quad (52)$$

Y el controlador de Matlab tipo PID tiene la estructura:

$$G(s) = K_p + K_i * \frac{1}{s} + K_D * \frac{N}{1 + N \frac{1}{s}} \quad (53)$$

Expresándolo en función del tiempo:

$$G(s) = K_p * e(t) + K_i * \int e(t) dt + K_D * \frac{de(t)}{dt} \quad (54)$$

Y para la implementación en Arduino sería de la forma discretizada:

$$G(s) = K_p * e[n] + K_i * e[k] * \tau + K_D * \frac{e[n] - e[n-1]}{\tau} \quad (55)$$

En donde, τ es el tiempo de muestreo, $e[n]$ es el error en el instante actual, $e[k]$ es el error total acumulado, $e[n-1]$ es el error en el instante anterior.

Descrito lo anterior, se muestra el lazo de control que se ha calibrado con los modelos de motores anteriormente calculados.

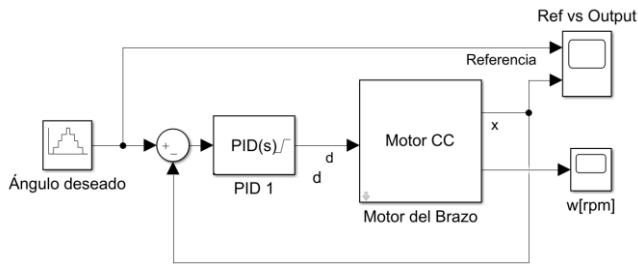


Fig. 58. Diagrama de Bloques del Controlador de los Motores Modelados

Para cada uno de los motores se ha obtenido, las siguientes constantes

TABLA XXIX. PARÁMETROS DE LOS CONTROLADORES DE LOS TRES MOTORES

Motor	Kp	Ki	kd
BRAZO-E192.24.36	20.62	0.1	0
CODO 378:1 25Dx58L 12V	11.62	0.1	0
ALTURA Y PINZA 1000:1 - HPCB 12V	2.62	0.1	0

Si la implementación se realiza en Arduino directamente se debe considerar la ecuación (55), en la que $K_i \cdot \tau = 0.1$, de lo cual, K_i depende del tiempo de muestreo que se determine, que para el robot se lo ha puesto en un valor de $\tau = 100 [ms]$, entonces el valor real de K_i a la hora de programarlo si se lo hace desde el IDE de Arduino es $K_i = 1$. Esto se verá reflejado en el código de programación anexo.

Los controles de posición para los motores presentan las siguientes curvas de posición en función del tiempo, para una referencia que varía de 0 a 360 grados en pasos de 90°

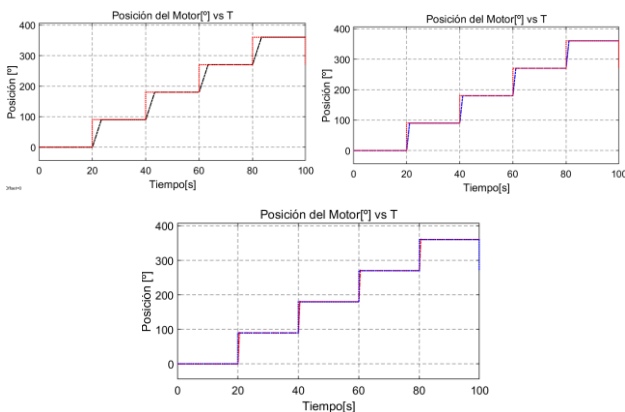


Fig. 59. Resultado del Control de posición de los 3 motores – Posición. Motor1, Motor2, Motor3

Y en cuanto al torque que ejerce el motor se tiene.

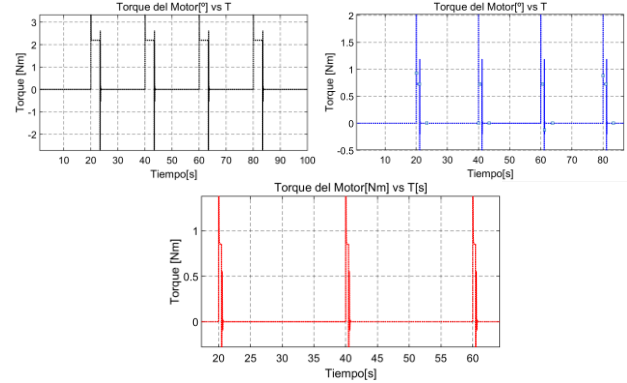


Fig. 60. Resultado del Control de posición de los 3 motores - Par. Motor1, Motor2, Motor3

Los picos que se ven son los transitorios, pero el valor en estado estable se detalla de manera resumida en la TABLA XXX.

TABLA XXX. RESULTADOS CONTROL DE POSICIÓN MOTORES

Motor	Tiempo de Establecimiento[s]	Erro de Posición	Tmax Alcanzado [Nm]
Brazo	3.41	0.08	2.2
Antebrazo	1.15	0.085	0.725
Altura y pinza	0.6	0.16	0.854

De esto se concluye que los controladores son adecuados, ya que tienen una buena respuesta en tiempo y error, su velocidad no supera los límites del motor, y el torque de los tres motores es inferior al máximo indicado en los datos de características.

XII. DISEÑO BASADO EN MODELO

Con los controladores diseñados ya puede pasar a su implementación sin embargo es conveniente en esta sección detenernos para realizar su simulación en tiempo real y ajustar parámetros si fuera necesario. El diseño basado en modelo que se presenta a continuación propone, hacer prueba de los controladores diseñados en el robot importado de SolidWorks a Simscape de Matlab, dicho modelo hace referencia exacta a las propiedades mecánicas del robot construido, por lo cual la interacción de masas e inercias se ve reflejada, entonces esto permite predecir el comportamiento real que presentará el mecanismo. Es cierto, que esta sección debería ser de las primeras, ya que antes de la construcción de un prototipo debe haber sido simulado y probado para poder predecir los cambios y mejoras antes de su construcción. Sin embargo, se lo describe en esta sección por seguir un orden y entendimiento en función del modelado y control de los motores.

El modelo importado en Simscape se puede ver como en la Fig. 61.

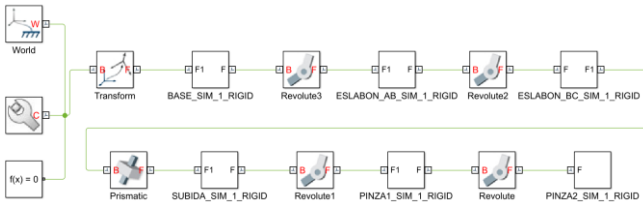


Fig. 61. Modelo del Robot Scara en Matlab - Simscape.

Se ha utilizado un modelo del Robot simplificado que contiene los datos del peso de engranes, tornillería, pero se lo ha importado de manera simplificada en subensambles con las relaciones de posición solo en los puntos de interés que es en donde interaccionan los motores.

La importación se la hace mediante un Plugin de Matlab para SolidWorks, y luego una vez el modelo importado se procede a la configuración de actuadores y sensores, es aquí en donde se ingresarán los modelos de controladores PID desarrollados en secciones previas.

El modelo ya con los controladores queda como en la Fig. 62

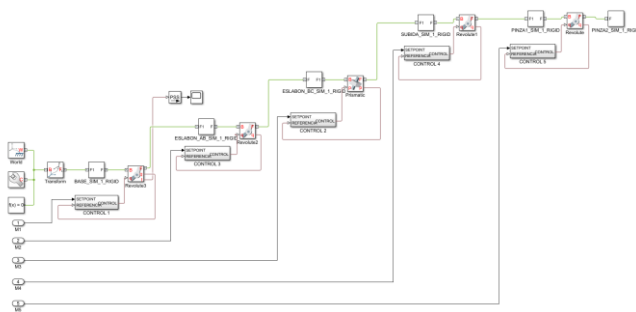


Fig. 62. Modelo del Robot Scara en Simscape – Con Controladores.

Los controladores tienen la forma de la Fig. 63 y contienen bloques de conversión de señales físicas a señales interpretables por Simulink

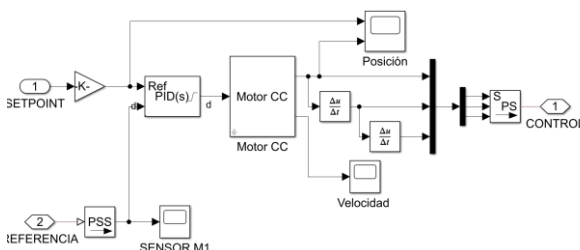


Fig. 63. Simscape – Diagrama de los controladores.

Finalmente, el Robot en el entorno de Simscape se puede ver como en la Fig. 64.

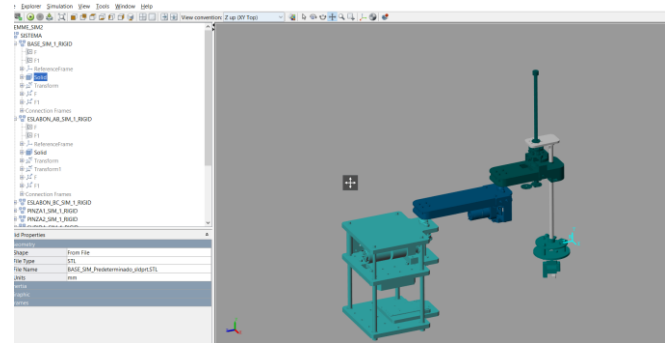


Fig. 64. Robot Scara – Simscape – Modelo 3D

Al realizar la simulación dinámica e implementar la cinemática inversa fue posible determinar que tanto en la realidad como en simulación se obtienen resultados similares. Se encontró que el motor del brazo es lento en relación con el motor del antebrazo, el motor del brazo va a una velocidad de 4.5 [rpm], mientras que el motor del antebrazo va a aproximadamente 14 [rpm], lo que hasta no haberlo visto en la simulación ni en la realidad no se había notado. Al realizar la cinemática inversa, y moverlo solo sobre el eje X o solo sobre el eje Y el robot tiende a realizar como una circunferencia para llegar de un punto a otra, en primer lugar esto se debe a que solo se está cambiando la referencia bruscamente, y en segundo lugar a que el motor 1 es lento, por lo que ambos motores no pueden compensar el movimiento a la vez y mientras el motor del antebrazo llegó a su posición, el motor 1 no ha podido llegar aún a su posición deseada, lo que hace que la trayectoria no sea exactamente recta.

XIII. IMPLEMENTACIÓN DE CONTROL INDIVIDUAL EN MATLAB - SIMULINK

Como se mencionó previamente, se hizo inicialmente la implementación de los controladores mediante la librería de Simulink para Arduino, y se mostrará brevemente la configuración realizada y los resultados obtenidos (Anexo). Esta forma inicialmente facilitó la programación de los controladores ya que se calibra el control PID con el motor modelado y se pasa el bloque al programa que tiene la programación de Arduino y se ejecuta.



Fig. 65. Proceso de funcionamiento con Simulink y Arduino

Para, ello se configura todo el código de Arduino en un S-function builder, en el caso del control de un Motor, la estructura en Simulink es como en la Fig. 66.

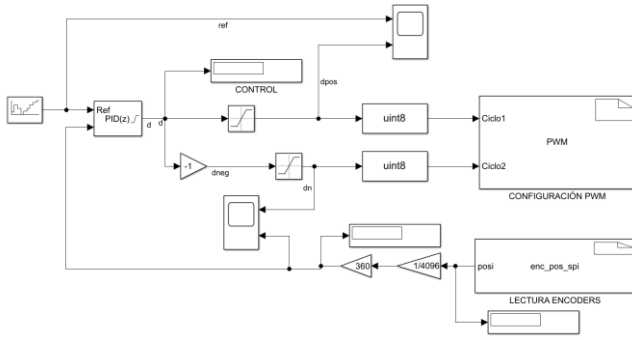


Fig. 66. Programación en Simulink para el control de Motores.

Este modelo contiene dos bloques de función para Arduino ya que uno de ellos configura las salidas de PWM del control y el otro es el bloque que contiene el código de la lectura de los encoders, como se explicó en la parte de electrónica utilizado, los encoders utilizan la comunicación SPI para el envío de sus datos, en este bloque se ha configurado para leer una n cantidad de encoders, ya que lo que difiere de la lectura de uno u otro encoders es la habilitación al estar en modo Maestro el Arduino y en modo Esclavo los encoders a leer.

El Bloque de función, básicamente consiste en las siguientes configuraciones.

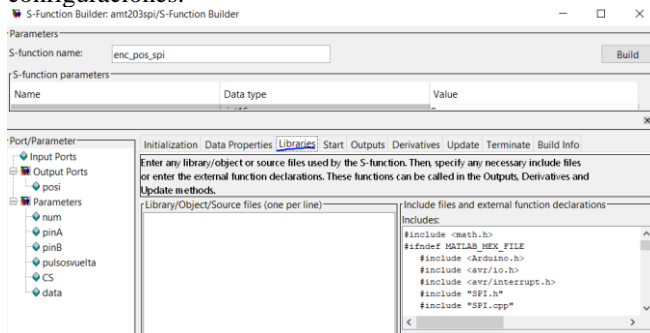


Fig. 67. S-function-Libraries- Configuración

Primero se configura la pestaña de Libraries, en la cual se hace el llamado y declaración de todas las librerías y la definición de las variables a utilizar, muy similar a la estructura de programación en el IDE de Arduino, en Matlab se lo va configurando en sus respectivas pestañas.

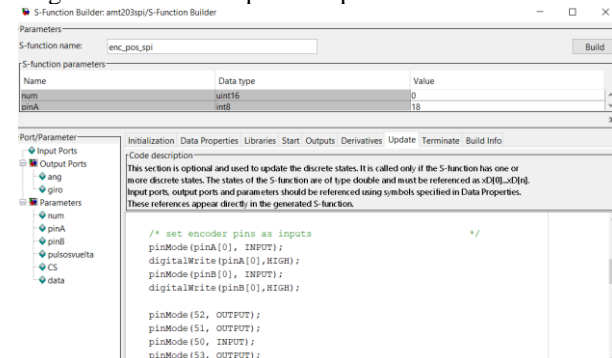


Fig. 68. S-function-Libraries- Update

En la pestaña Update se configura el Setup de Arduino, se declaran todos los pines que serán entradas o salidas, o se activan las respectivas interrupciones, etc.

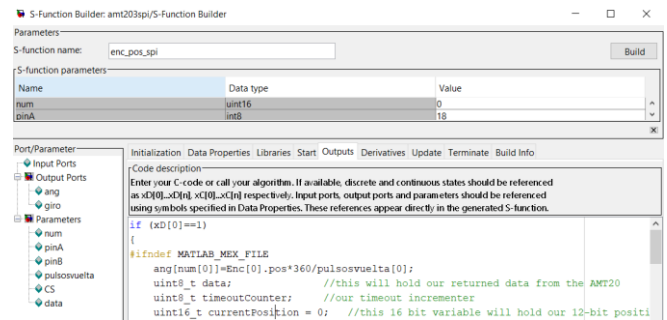


Fig. 69. S-function-Libraries- Outputs

En la pestaña de Outputs se configura el Loop para el Arduino, se puede ver que tiene la línea de código: #ifndef MATLAB_MEX_FILE la cual indica que esto se ejecuta fuera de Matlab y corresponde a lo que Arduino debe procesar.

Y cuando se ha ingresado el código, Matlab genera un sinnúmero de archivos de código, entre ellos el archivo __wrap-per.cpp que Matlab lo genera con código de C++, el cual requiere editar su código y agregar lo siguiente: extern "C" para que sea interpretable como código de C, además se debe considerar que Matlab no tiene todas las librerías de Arduino, para lo cual, se deben agregar a la carpeta en que se ejecuta el programa.

Todo lo anterior se ha mencionado con el fin de clarificar que Matlab genera varios programas los cuales no son modificables por el usuario y hacen que el proceso de edición de programa o búsqueda de error se haga mucho más lento que si se tuviera escrito línea por línea y se pueda hacer Debug.

Entonces, una vez generados los códigos se lo carga al Arduino y los resultados son los siguientes para el caso del control de un Motor Individual.

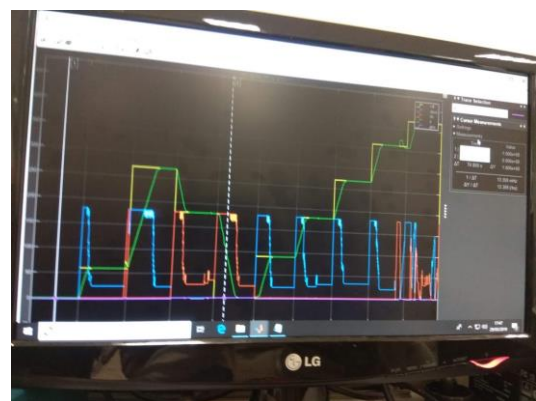


Fig. 70. Control de un Motor individual mediante Simulink

Los cuatro controladores fueron probados en Simulink de forma individual (Se puede ver en los anexos los videos de funcionamiento), sin embargo debido a que el código no es transparente para nosotros como usuarios y que luego se necesitaba incluir muchos más requerimientos como la cinemática inversa, interrupciones de cambio de mando, o los cuatro controladores al mismo tiempo y funcionando en tiempo real decidimos optar por programar directamente en el IDE de Ar-



duino y comunicarnos de la misma forma con la que se comunica Arduino con Matlab: Comunicación Serial. Entonces es allí en donde viene la siguiente etapa de la implementación que no deja de ser **Software in the loop** ya que si queremos inclusive podemos estar modificando las constantes del controlador en tiempo real sin problema alguno.

XIV. IMPLEMENTACIÓN DE LOS CONTROLADORES EN EL IDE DE ARDUINO

Después de hacer la transición de Simulink-Arduino, y optar por una solución más clara y visible para nosotros como programadores del Robot, en esta aplicación particular; pasamos a la implementación mediante programación normal en el IDE de Arduino. Entonces, el diagrama general de funcionamiento es el siguiente:

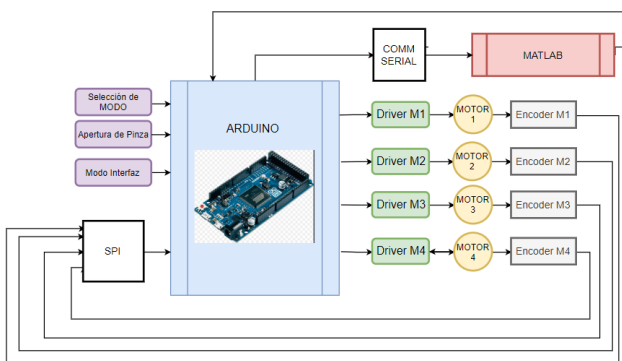


Fig. 71. Diagrama general de Arduino.

Con esta introducción de funcionamiento el programa realizado en el IDE de Arduino tiene el siguiente diagrama de flujo.

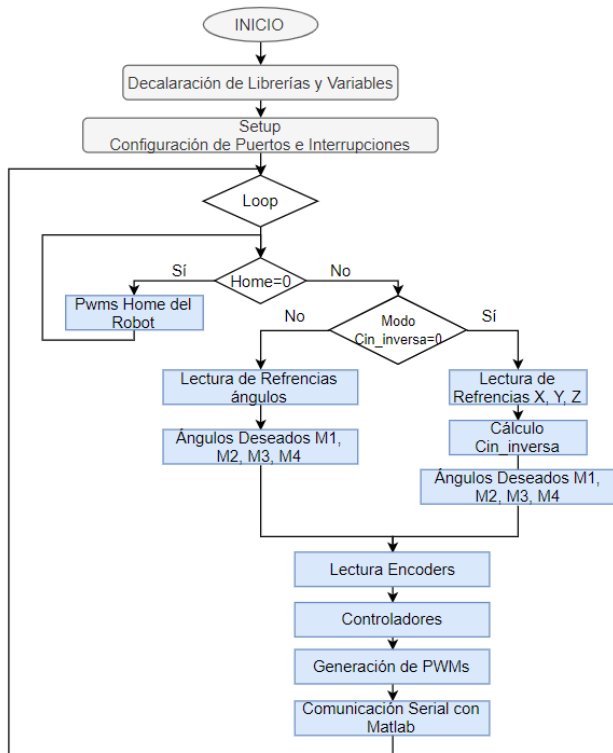


Fig. 72. Programación en Simulink para el control de Motores.

Como se puede ver el programa sigue una estructura lógica y clara. Además, en esta sección es conveniente mencionar que los controladores fueron programados de forma discretizada como se señaló en la ecuación (55), y que sus coeficientes son los mencionados en la TABLA XXIX.

Posterior a esta etapa, viene la etapa de la visualización, se utilizó para esto dos plataformas de Software:

Primero Matlab, que se encarga de toda la comunicación dúplex con Arduino y luego VREP (Virtual Robot Experimentation Platform) que se encarga de la visualización en 3D de nuestro prototipo.

En función de esto, se va a detallar a continuación cada una de las partes.

A. Interfaz en Matlab

Tiene la siguiente ventana de Inicio



Fig. 73. Ventana de Inicio - Interfaz

Y luego una ventana de visualización de datos y envío de referencias y comandos.

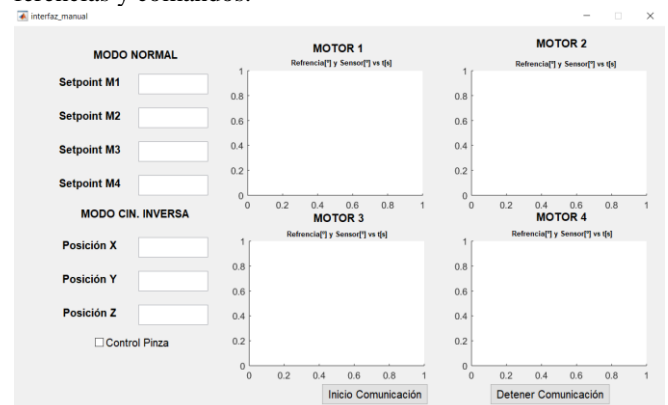


Fig. 74. Ventana de Comandos - Interfaz

Es necesario aclarar que, inicialmente se propuso trabajar con App Designer, sin embargo, se migró a Guide ya que facilita la comunicación serial y es más rápido en la transmisión de datos.



B. Robot en VREP

VREP se ha comunicado con Matlab en tiempo real mediante Api, el robot desarrollado en SolidWorks ha sido importado a VREP y configurado para moverse en función de los datos enviados por Matlab, que son los datos en tiempo real de lo que está pasando en la realidad con nuestro Robot Scara.

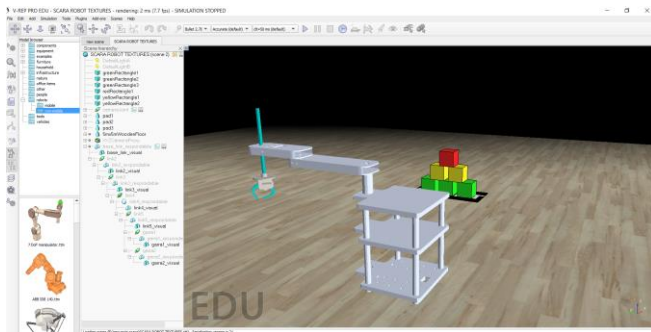


Fig. 75. Robot Scara en VREP

Con todo lo anterior el robot es capaz de funcionar en tiempo real y visualizar en VREP cada uno de sus movimientos, ya que los datos que obtiene VREP son los datos que los encoders tienen en ese momento. Es conveniente indicar que si movemos el Robot desde VREP no se podrá mover en la realidad ya que está funcionando solo en modo Receptor de datos.

XV. CONCLUSIONES

Tanto el diseño de los componentes estructurales como los componentes adquiridos comercialmente son adecuados, y en la mayoría sobredimensionados, para las cargas que se le van a aplicar al robot.

El montaje final y el acceso a áreas de atornillado resultó sencillo debido al montaje modular ideado y el diseño realizado.

Para la implementación de un controlador, es necesario tener un modelado adecuado del sistema a controlar, y se pudo comprobar que en base al modelo de los motores simulados en Simulink se programó en Arduino y las respuestas son tal como se esperaba y presentan una respuesta muy parecida o similar a la simulación

Se debe tener en cuenta en el diseño de una placa electrónica el ancho mínimo de pista, ya que una pista muy delgada al momento de su realización de manera manual puede desaparecer y perder el camino de conducción de cobre

La programación en un Software está dada por los beneficios que presentan estos hacia los usuarios, pudimos constatar que Matlab en una fase inicial fue una ayuda para la programación de los controladores, sin embargo, a la hora de incorporar un mayor número de sistemas y entradas o salidas se vuelve complicado su entendimiento, por lo que un Software que permita el acceso transparente a todo el código es lo más deseable.

Cuando se realiza una placa electrónica se debe considerar su aislamiento ante el ruido, ya que se ha comprobado en el diseño de nuestro robot que el ruido generado por los motores y la fuente de tensión afecta enormemente a las interrupciones que ingresan al microcontrolador, esto fue posible de solventar con una menor tensión de funcionamiento o a su vez es deseable conectar a tierra todos los elementos metálicos que conforman el Robot.

La simulación basada en modelo es sumamente importante ya que permite tener en simulación un prototipo sin necesidad de haberlo construido y determinar posibles problemas que se puede tener a futuro, por ejemplo, se puede calibrar en simulación los controladores ya que el modelo simulado conserva todas las características físicas que tiene el prototipo real, entonces se puede analizar posibles mejoras a partir de la simulación

XVI. ANEXOS

TABLA XXXI. RELACIÓN DE ANEXOS ENTREGADOS

Descripción del Anexo	Nombre del Archivo
Listado de señales de la tarjeta de control	Listado_señales.xlsx
Datasheet microcontrolador SAM3X8E ARM Cortex-M3 de 32-bits	AtmelSAM3X_Microcontroller_Datasheet.pdf
Datasheet cojinetes articulados c/ soporte de EFOM	igubal EFOM.pdf
Datasheet casquillo con brida Iglidur P Ref PFM-1012-07	iglidur_P.pdf
Driver para motores DC TB9051FTG	Driver_TB9051FTG.pdf
Datasheet encoder AMT	Encoder_amt20.pdf
Listado de piezas y componentes pedidos	Piezas y Componentes.xlsx
Diagrama de Gantt del proyecto	GANTT_PROYECTO SCARA_SEM.pdf
Diagrama de flujo del programa de control desarrollado en Arduino	Diagrama_flujo_arduino.pdf
Planos del robot y fabricación de piezas	RG3.00.pdf
Síntesis dimensional de robot SCARA	Síntesis dimensional de robot SCARA.pdf
Modelado, simulación y control de manipulador SCARA	Modelado, simulación y control de manipulador SCARA (V2.0).pdf
Código desarrollado para la plataforma Arduino	Cod_Arduino_G3.ino
Diseño del circuito de control y PCB del Shield desarrollado	Circuito_control_SCARA_G3.pdspd
Video de Funcionamiento en Simscape	Simscape_SacaraG3.mp4
Archivo del Scara en VREP	SCARA ROBOT G3 VREP.ttt
Código de Arduino	Cod_Arduino_G3.ino
Interfaz	interfaz_manual.m / interfaz_manual.fig Inicio.m / Inicio.fig
Modelado de Motores en Matlab	Modelado_de_Motores.slx
Calibración de Controladores en Matlab	Controladores_de_Motores.slx
Modelo en Simscape	Simscape_G3.slx



XVII. REFERENCIAS

- [1] P. Corke, «robotacademy.net,» [En línea]. Available: <https://robotacademy.net.au/lesson/inverse-kinematics-for-a-2-joint-robot-arm-using-geometry/>.
- [2] M. G. Quijano, «Obtención Experimental de los parámetros del Motor,» Universidad Pontificia Bolivariana, 2009.
- [3] A. Noriega, «Modelado y simulación cinemática».
- [4] A. Noriega, «Modelado, simulación y control de manipulador SCARA».
- [5] A. Noriega, «Síntesis dimensional de robot SCARA».