

UNIVERSIDAD NACIONAL DE INGENIERÍA
FACULTAD DE INGENIERÍA MECÁNICA
ESCUELA PROFESIONAL DE INGENIERÍA MECATRÓNICA



ANÁLISIS Y CONTROL DE ROBOTS

MT517

**Informe 3: Modelamiento y Control de un
Manipulador Robótico de 2 Grados de Libertad**

ALDAVE TORRES, Carlos Felipe Sebastián - 20200159H

CASTRO SUAZO, Fidel Angel - 20200062D

CONTRERAS AVENDAÑO, Juan Sebastian Francesco - 20202024B

LLONTOP HERRERA, Marco Leandro - 20202011H

Revisado por

Ing. Calle Flores Iván

27 de Noviembre de 2023

Resumen

En este laboratorio, los estudiantes, en base de las características transitorias de las variables de estado, la controlabilidad del sistema y otras consideraciones prácticas y teóricas, se diseñará e implementará un controlador de estados. Se diseñarán distintos controladores para luego justificar el sistema controlado adecuado a implementar. Este diseño final será validado a través del osciloscopio y apoyado con la simulación en Matlab, Simulink Y proteus.

Además, se simulará un observador de orden completo y de orden reducido para luego obtener compensadores basados en estos observadores y en el controlador diseñado.

Índice general

1.	Implementación de la Electrónica	2
1.1.	Implementación del sistema electrónico	2
1.2.	Análisis del comportamiento del driver	6
2.	Modelamiento del Sistema Dinámico	7
2.1.	Parámetros dinámicos de cada motor DC	7
2.2.	Ecuaciones de cinemática directa	19
2.3.	Centros de masa de eslabones y motores	21
2.4.	Jacobianos de eslabones y motores	23
2.5.	Matrices de inercia de los eslabones y motores	25
2.6.	Modelo dinámico	26
2.7.	Simulación del modelo dinámico	28
2.8.	Validación de los motores	35
3.	Control descentralizado	36
3.1.	Ánálisis de resultados para diferentes configuraciones	36
3.2.	Código ingresar variables deseadas en el espacio X	38
4.	Anexos	40
4.1.	Mediciones de masa	40

1. Implementación de la Electrónica

1.1. Implementación del sistema electrónico

A continuación se presenta el esquema de conexiones del driver.

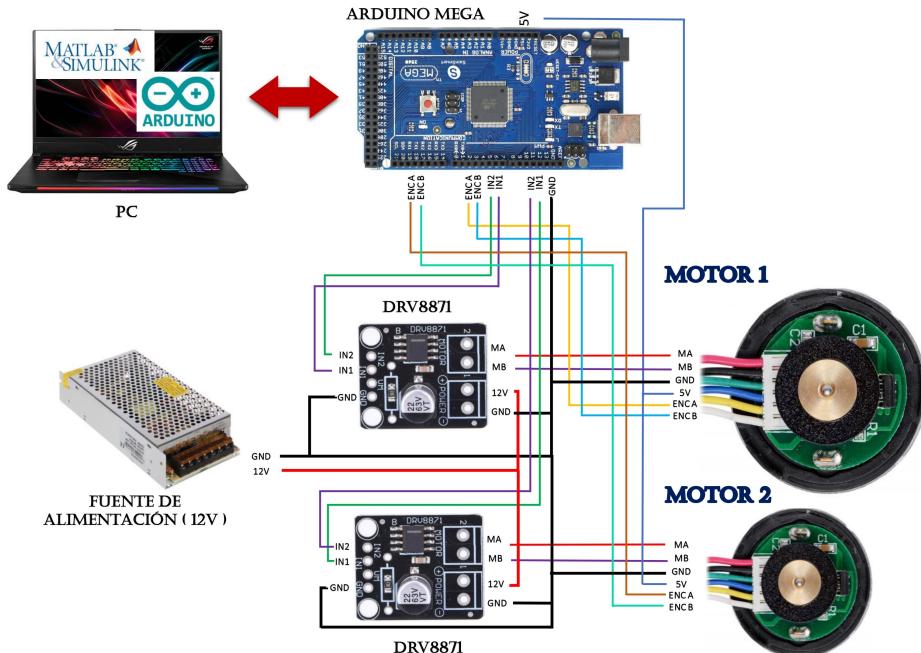


Figura 1.1: Conexión del driver DRV8871

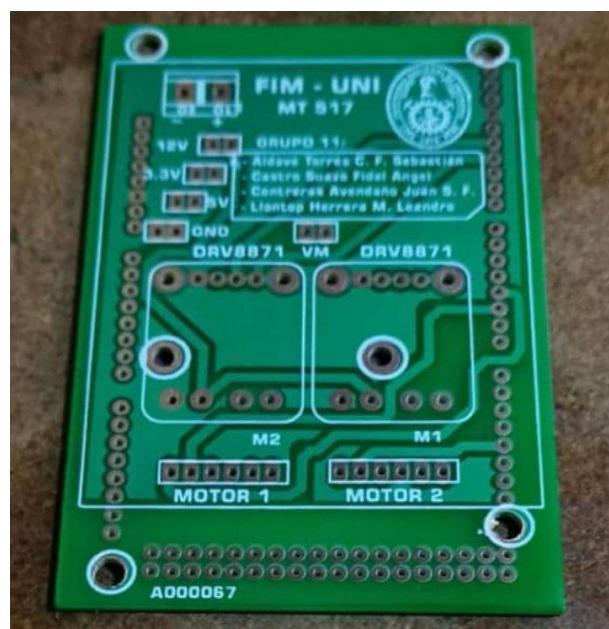


Figura 1.2: PCB

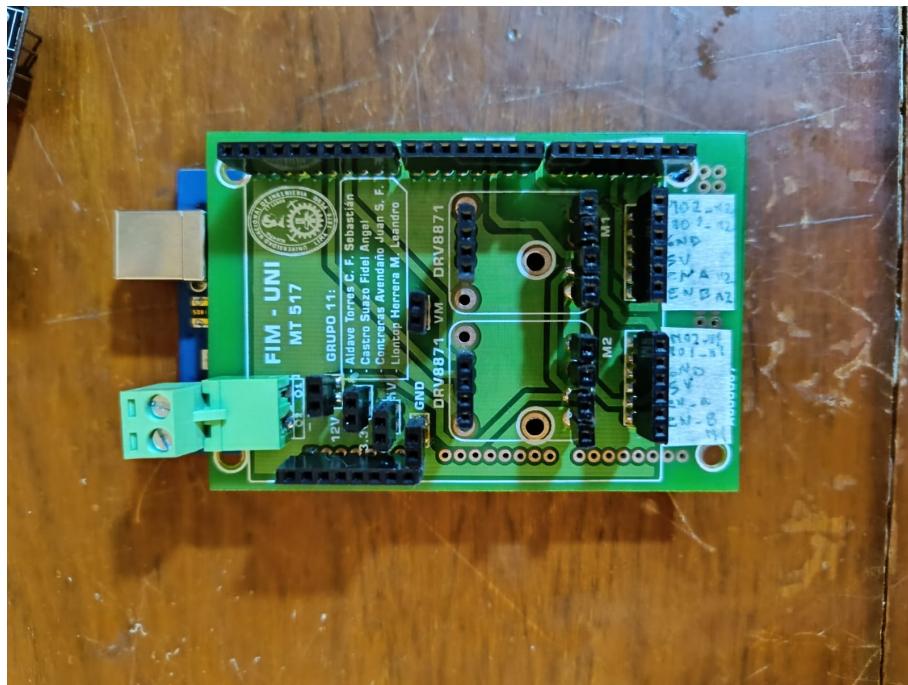


Figura 1.3: Shield con los espadines soldados y conectado al Arduino Mega

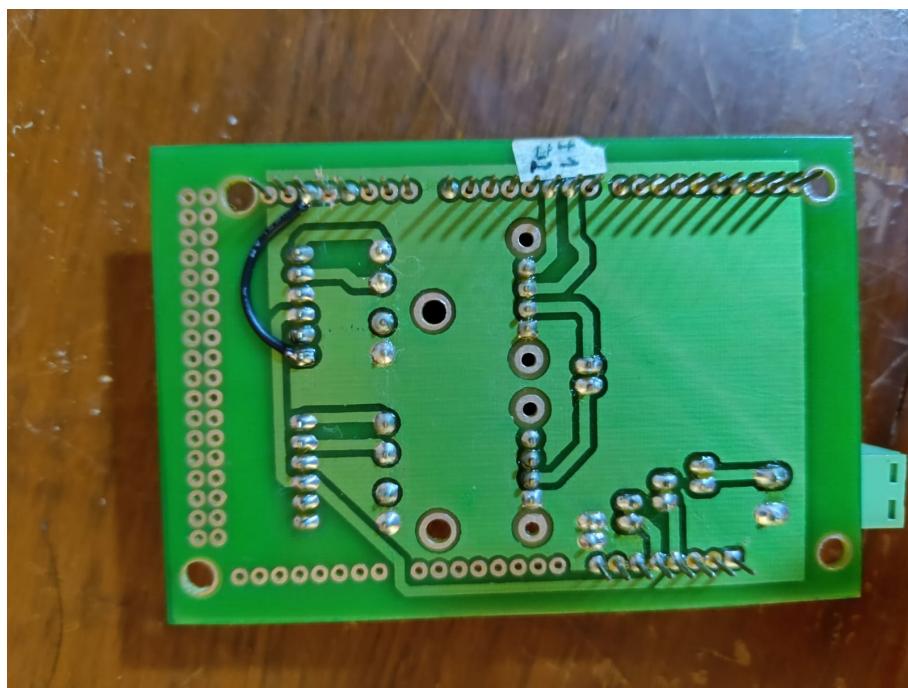


Figura 1.4: Reverso del shield

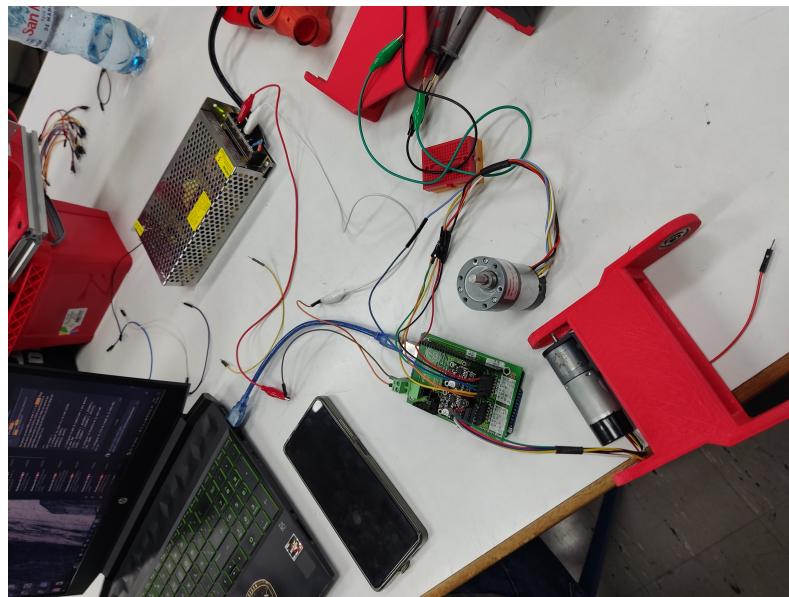


Figura 1.5: Implementación con el shield

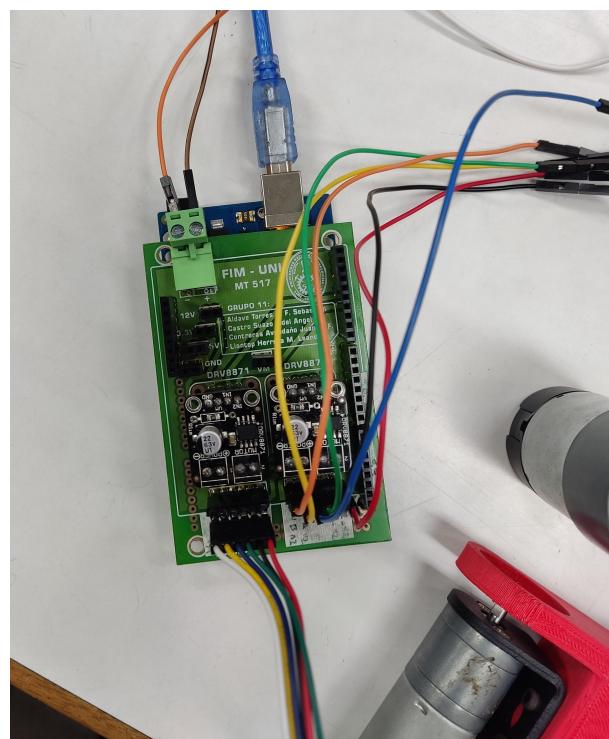


Figura 1.6: Implementación con el shield

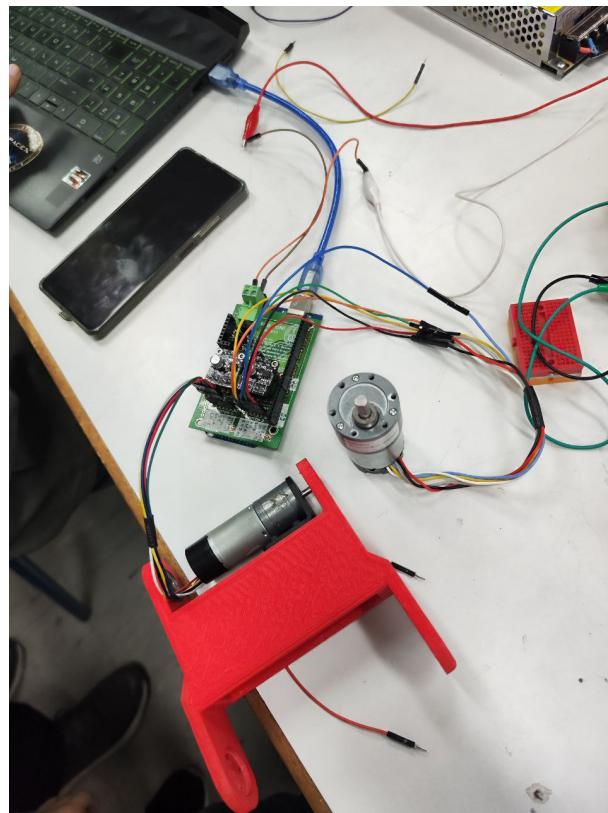


Figura 1.7: Conexión de ambos motores

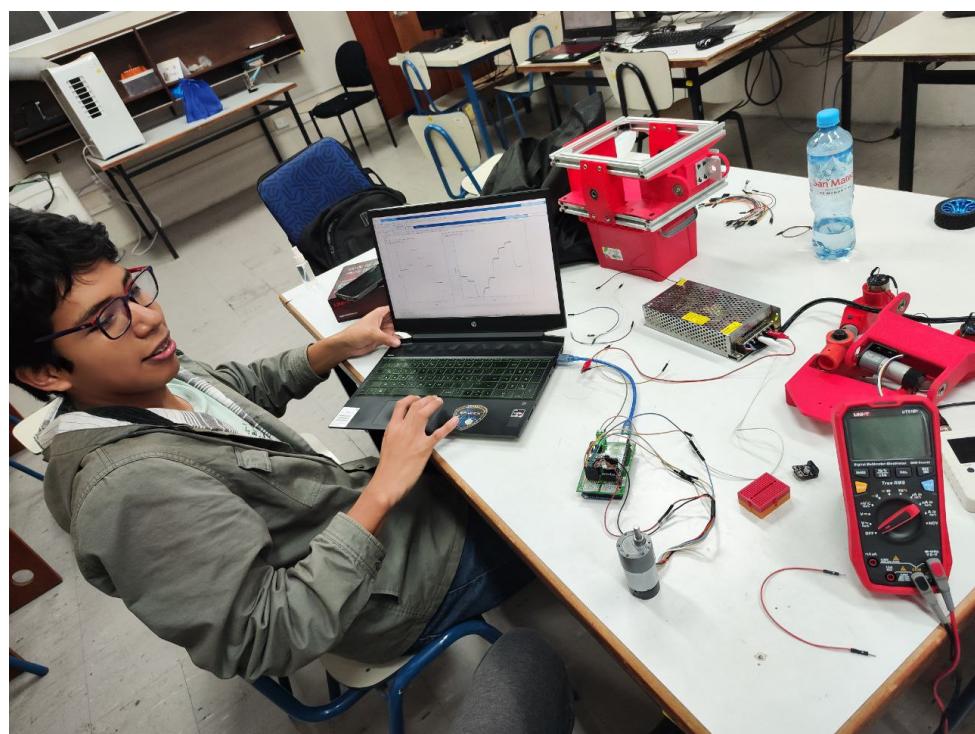


Figura 1.8: Pruebas con el motor

1.2. Análisis del comportamiento del driver

Se analizó el comportamiento del driver DRV8871, con el motor conectado, para hallar la ganancia y verificar la linealidad, el driver se configuró en modo drive/brake.

PWM	Vout	Vin	Vout/Vin
10	0.06	0.048	1.25
20	0.18	0.097	1.8556701
30	0.28	0.146	1.9178082
40	0.31	0.195	1.5897436
50	0.36	0.244	1.4754098
60	0.47	0.293	1.6040956
70	0.54	0.341	1.5835777
80	0.8	0.39	2.0512821
90	0.91	0.439	2.0728929
100	1.03	0.488	2.1106557
150	1.6	0.733	2.1828104
160	1.71	0.781	2.1895006
200	2.17	0.977	2.221085
250	2.73	1.221	2.2358722
300	3.29	1.465	2.2457338
350	3.85	1.711	2.2501461
400	4.41	1.955	2.2557545
450	4.98	2.2	2.2636364
500	5.55	2.443	2.271797
600	6.66	2.932	2.271487
700	7.77	3.421	2.2712657
800	8.88	3.91	2.2710997
900	9.98	4.397	2.2697294
1000	11.11	4.886	2.2738436
1023	11.39	4.999	2.2784557
			2.0838897

Figura 1.9: Recolección de datos con una frecuencia de 7.82 kHz

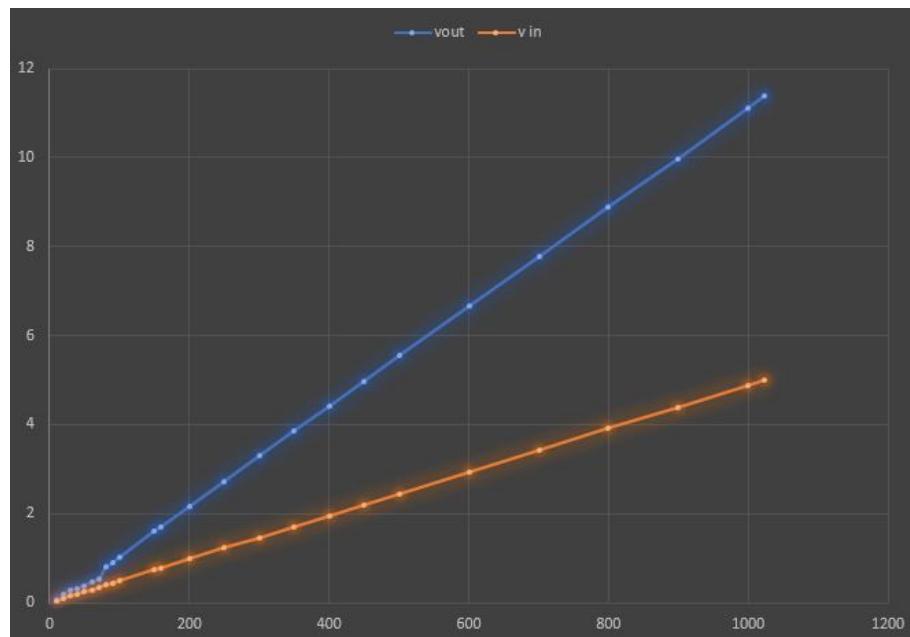


Figura 1.10: Vout vs PWM, Vin vs PWM

2. Modelamiento del Sistema Dinámico

2.1. Parámetros dinámicos de cada motor DC

El modelo de un motor DC se puede dividir en dos partes: eléctrica y mecánica.

Parte eléctrica de un motor DC

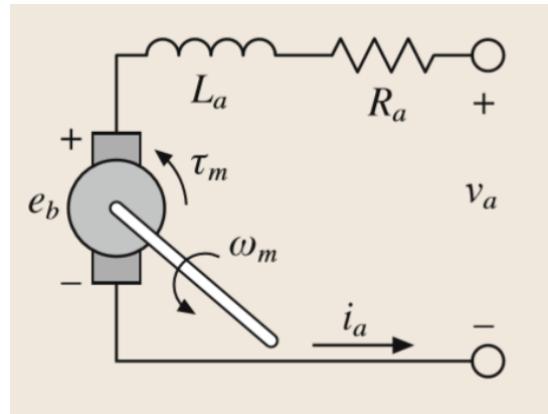


Figura 2.1: Circuito de armadura de un motor DC

$$v_a = R_a i_a + L_a \frac{di_a}{dt} + v_b \quad v_b = K_b \omega_m$$

donde R_a y L_a son respectivamente la resistencia (Ω) y la inductancia de la armadura (H), y K_b es la constante del voltaje contraelectromotriz ($V \cdot s/rad$).

Parte mecánica de un motor DC

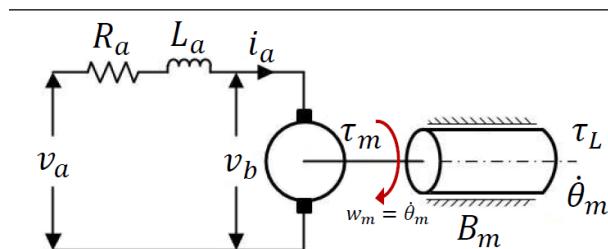


Figura 2.2: Balance de torques de un motor DC

$$\tau_m = J_m \frac{d\omega_m}{dt} + B_m \omega_m + \tau_L \quad \tau_m = K_t i_a$$

donde J_m y B_m representan el momento de inercia ($kg \cdot m^2$) y coeficiente de fricción viscosa ($kg \cdot m^2/s$) en el eje del motor.

Para hallar los parámetros dinámicos del **motor reductor JGA25-370 12V de 77RPM** se siguen los siguientes pasos:

- **Resistencia de armadura:** En principio, se puede medir la resistencia simplemente con un multímetro:

$$R_a(\text{multímetro}) = 7,2 \Omega$$

Otro forma más exacta es usar PWM (10 bits) y un voltaje de 5V para modular pequeños valores del voltaje de armadura y durante las mediciones se mantiene la corriente constante. Todo ello con el fin de conseguir $\frac{di_a}{dt} = 0$ y $\omega_m = 0$ para así obtener la siguiente relación en el circuito de armadura:

$$v_a = i_a R_a$$

Midiendo el voltaje y corriente de armadura se podrá hallar la resistencia por medio de la relación anterior.

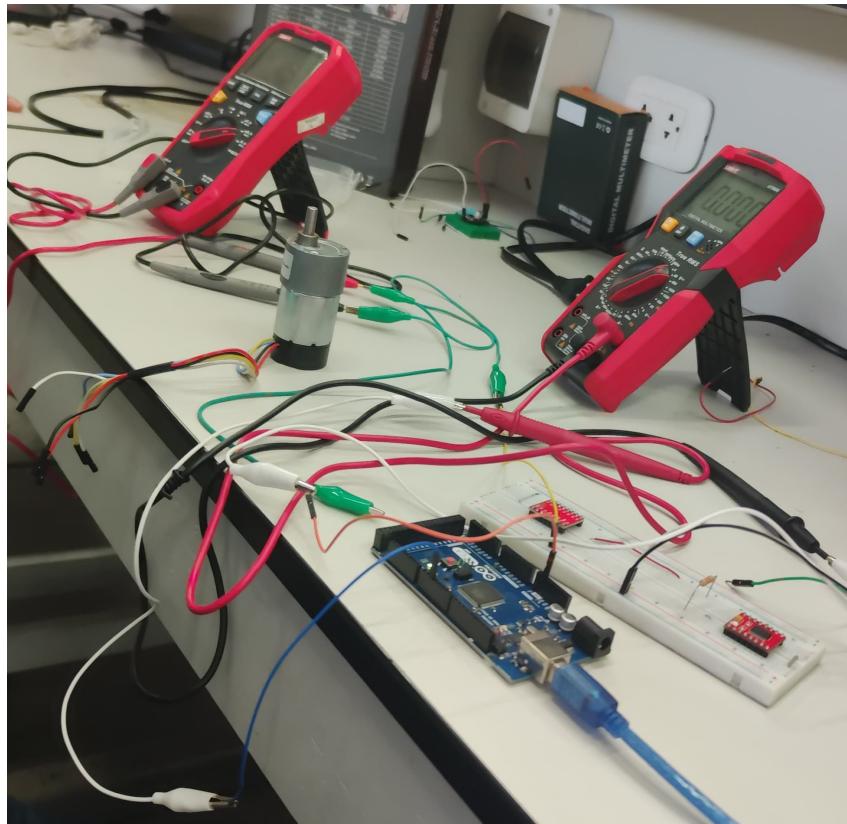


Figura 2.3: Mediciones de corrientes y voltajes

PWM	Voltaje (mV)	Ia (mA)	w (rad/s)	Resistencia (Ohm)
0	0	0	0	-
2	2.7	0.29	0	9.310344828
4	4.2	0.45	0	9.333333333
6	5.9	0.63	0	9.365079365
8	7.5	0.82	0	9.146341463
10	9.2	1.02	0	9.019607843
12	10.8	1.23	0	8.780487805
14	12.5	1.45	0	8.620689655
16	14.2	1.69	0	8.402366864
18	16.1	1.92	0	8.385416667
20	18.2	2.17	0	8.387096774
22	20.3	2.43	0	8.353909465
24	22.7	2.7	0	8.407407407
26	25.1	2.97	0	8.451178451
28	27.6	3.24	0	8.518518519
30	30.3	3.52	0	8.607954545
32	33	3.81	0	8.661417323
34	35.8	4.1	0	8.731707317
36	38.6	4.39	0	8.792710706
38	41.5	4.69	0	8.848614072
40	44.4	4.98	0	8.915662651
				8.751992253

Figura 2.4: Múltiples mediciones con variables pertinentes: $R_a = 8,751992253 \Omega$

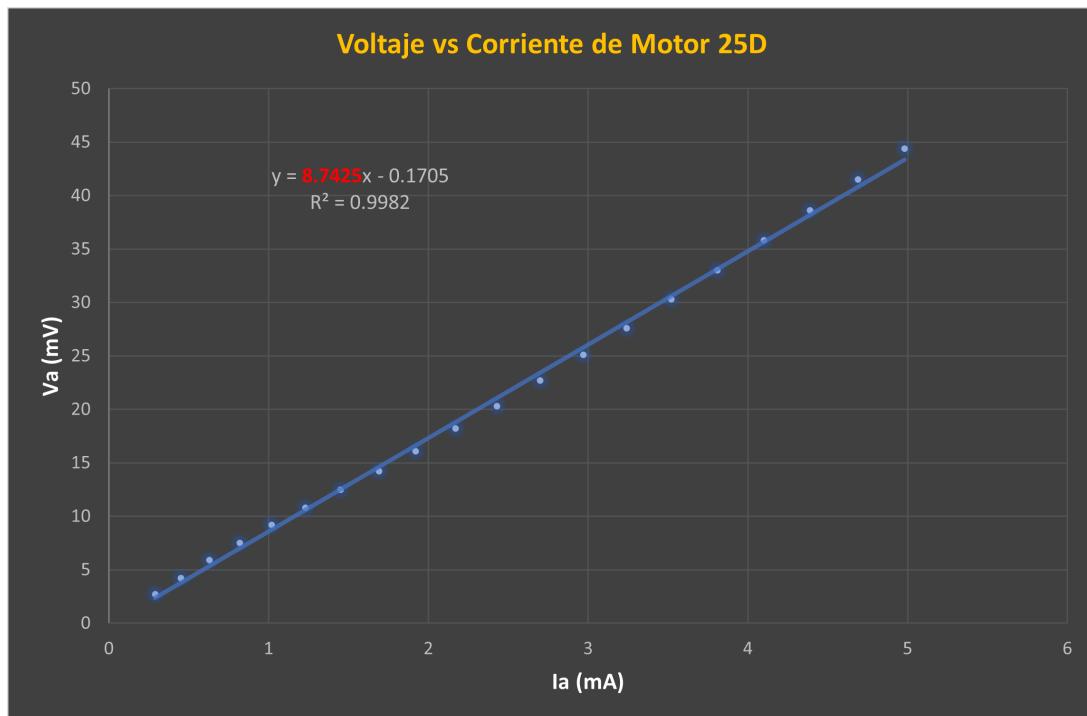


Figura 2.5: Verificación de la linealidad de la relación de voltaje y corriente

- **Inductancia de armadura:** Al circuito de armadura se añade una resistencia en serie para tener valores pequeños de corriente para conseguir $\omega_m = 0$ y, en consecuencia, $v_b = 0$. De esta manera, se tendrá un circuito RL del cual se puede obtener la constante de tiempo al aplicar una señal cuadrada de 5 V de amplitud y 2 kHz



Figura 2.6: Medición de la resistencia en serie usada: $R_{serie} = 326,35 \Omega$

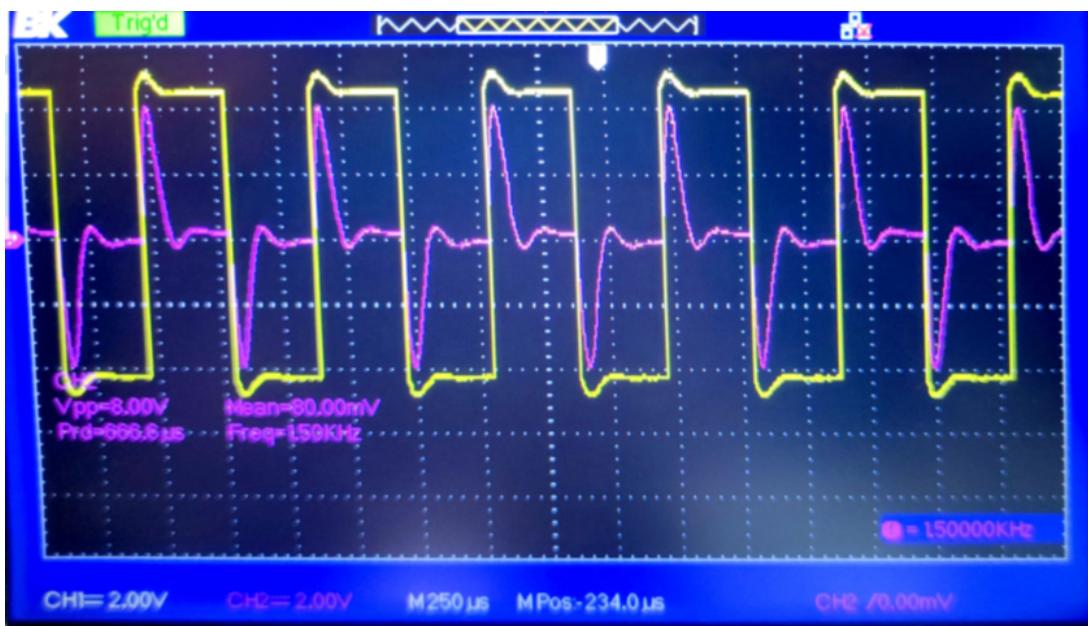


Figura 2.7: Señal cuadrada de entrada y voltaje de respuesta en R

Se usa el criterio del 37 % del voltaje máximo para hallar la constante de tiempo:

$$V_2 = 37\%V_{max} = 0,37 \times 2,11 = 0,7807 V$$

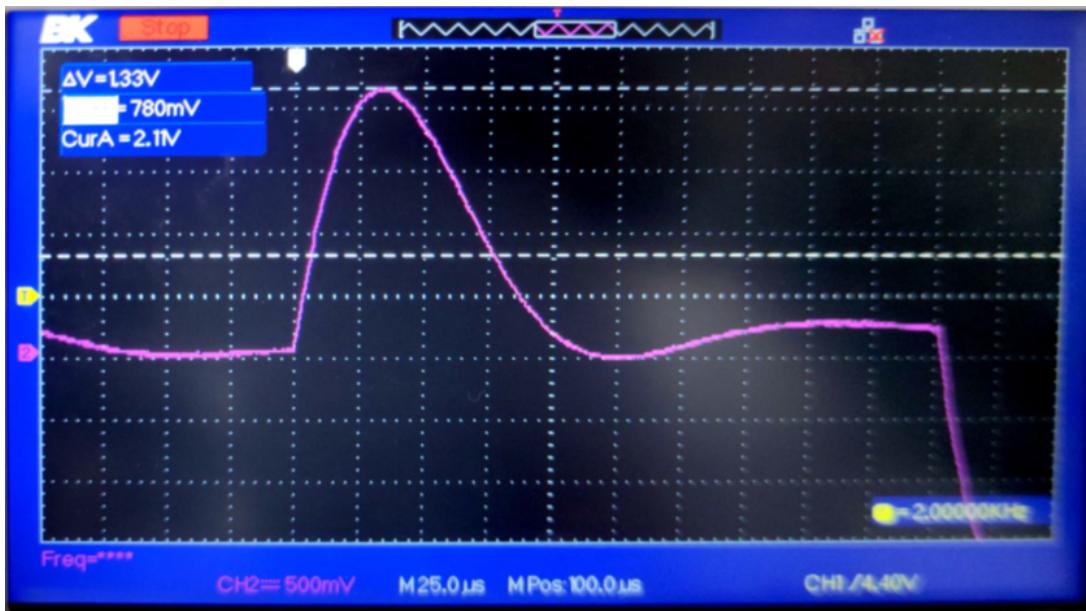


Figura 2.8: Identificación de voltajes entre los que se medirá el tiempo

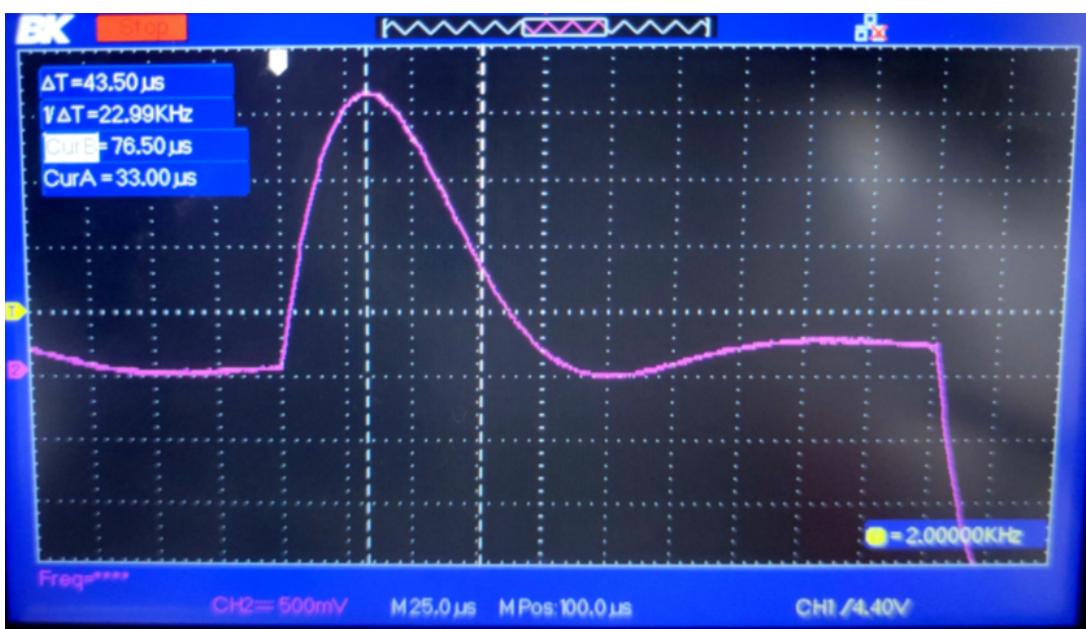


Figura 2.9: Medición de la constante de tiempo: $\tau = 43,5\ \mu s$

La ecuación de constante de tiempo para un circuito RL permite calcular:

$$\tau = \frac{L_a}{R} \rightarrow L_a = \tau(R_a + R_{serie}) \rightarrow L_a = 14,57693666\ \text{mH}$$

- **Estimación de parámetros mediante Simulink:** Se envía un patrón de voltajes al motor y se recepciona las señales del encoder para hallar la velocidad en cada caso. Tener en cuenta que $CPR = 78,09 \times 48 = 3748,32$ cuentas por revolución.

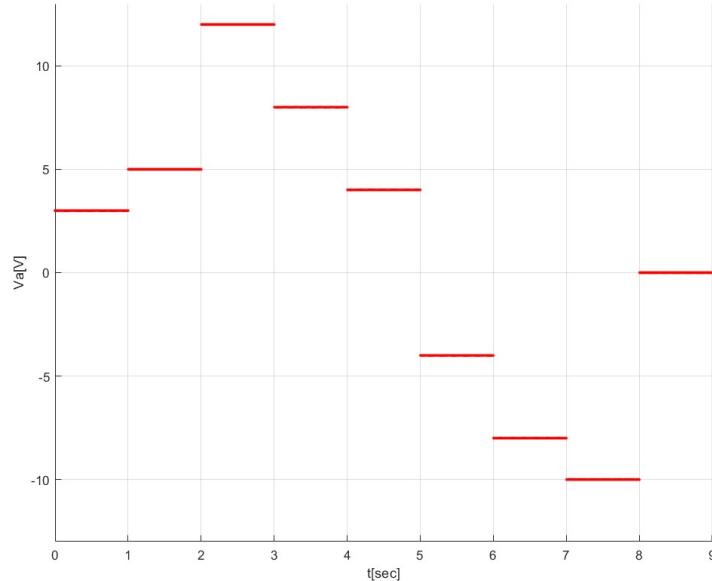


Figura 2.10: Patrón de voltajes enviados al motor mediante PWM de 10 bits

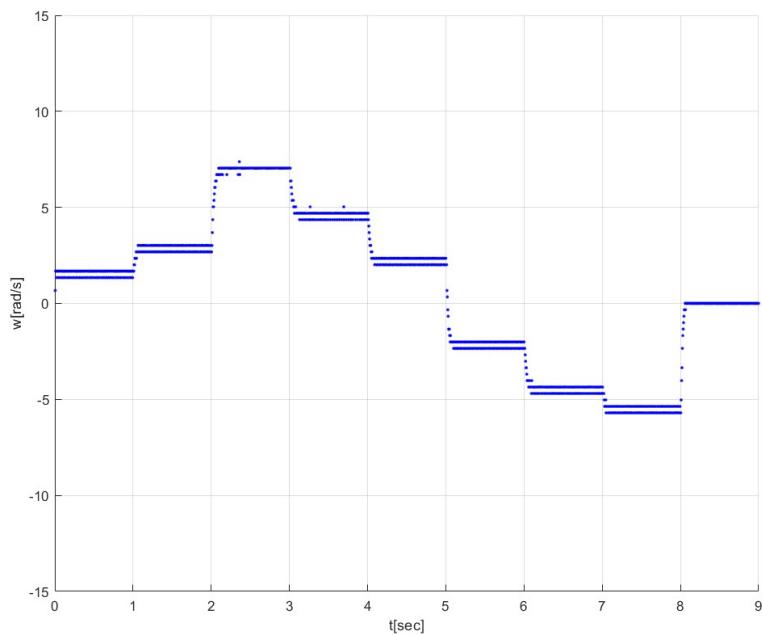


Figura 2.11: Data de velocidad angular obtenida mediante Arduino y Matlab

Obtenidos los datos de voltaje y velocidad angular, mediante el modelo del motor hecho en Simulink y con los datos de resistencia e inductancia de armadura se usa la herramienta *Parameter Estimation* para hallar los parámetros faltantes del modelo.

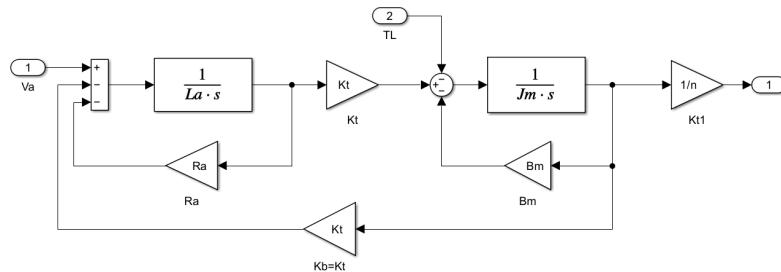


Figura 2.12: Modelo del motor en Simulink

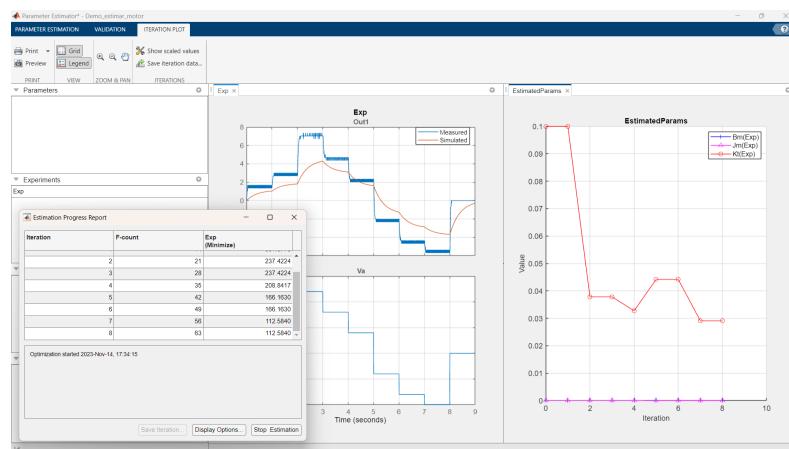


Figura 2.13: Iteraciones mediante *Parameter Estimation*

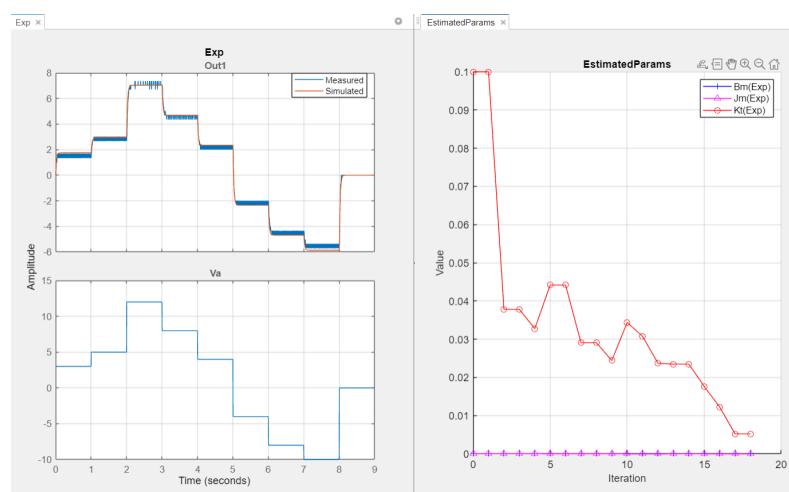


Figura 2.14: Convergencia de la estimación: parámetros hallados

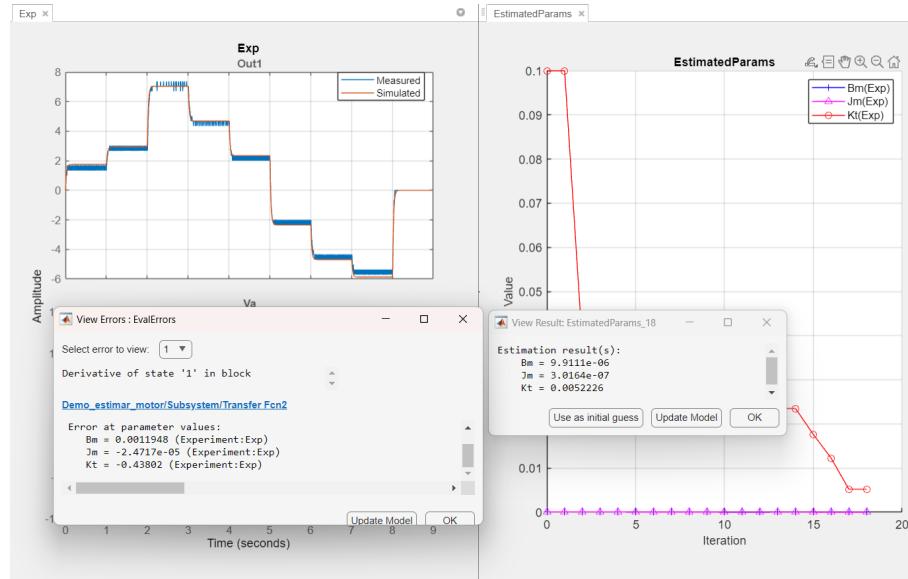


Figura 2.15: Parámetros que ajustan el modelo a la data de velocidad angular

Para motores balanceados, se puede afirmar que $K_t = K_b$. Entonces:

$$B_m = 9,9111 \times 10^{-6}$$

$$J_m = 3,0164 \times 10^{-7}$$

$$K_t = K_b = 0,0052226$$

- Validación de los parámetros mediante el modelo en espacio estado continuo:

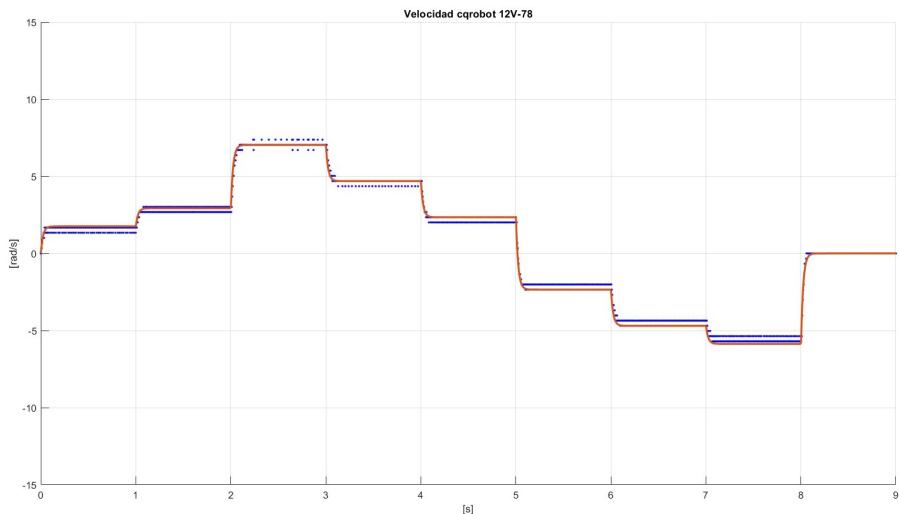


Figura 2.16: Validación de parámetros comparando modelo en espacio estado y la data

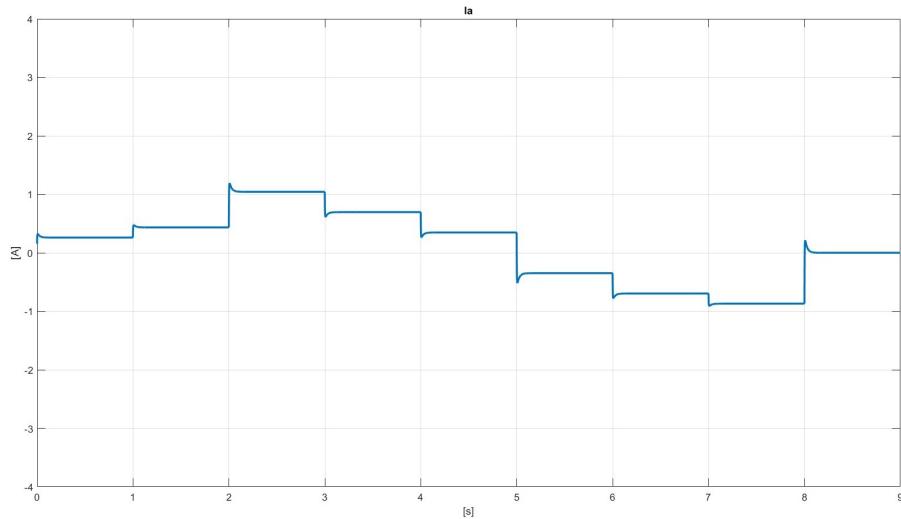


Figura 2.17: Corriente de armadura en respuesta al patrón de voltaje aplicado

Se sigue el mismo procedimiento para hallar los parámetros dinámicos del **motor reduedor metálico 37D de 150 rpm**:

- Resistencia de armadura:

$$R_a(\text{multímetro}) = 2,7 \Omega$$

PWM	Voltaje (mV)	I_a (mA)	w (rad/s)	Resistencia (Ohm)
0			0	-
2	1.6	0.34	0	4.705882353
4	2.8	0.54	0	5.185185185
6	3.6	0.75	0	4.8
8	4.3	0.98	0	4.387755102
10	5.4	1.22	0	4.426229508
12	6.7	1.47	0	4.557823129
14	8	1.73	0	4.624277457
16	9.5	2.01	0	4.726368159
18	11	2.3	0	4.782608696
20	12.6	2.6	0	4.846153846
22	14.2	2.9	0	4.896551724
24	16	3.22	0	4.968944099
26	17.7	3.54	0	5
28	19.5	3.86	0	5.051813472
30	21.3	4.2	0	5.071428571
32	23.3	4.53	0	5.143487859
34	25.3	4.87	0	5.195071869
36	27.8	5.21	0	5.335892514
38	30.8	5.56	0	5.539568345
40	34.2	5.9	0	5.796610169
				4.952082603

Figura 2.18: Promedio de resistencias: $R_a = 4,952082603 \Omega$

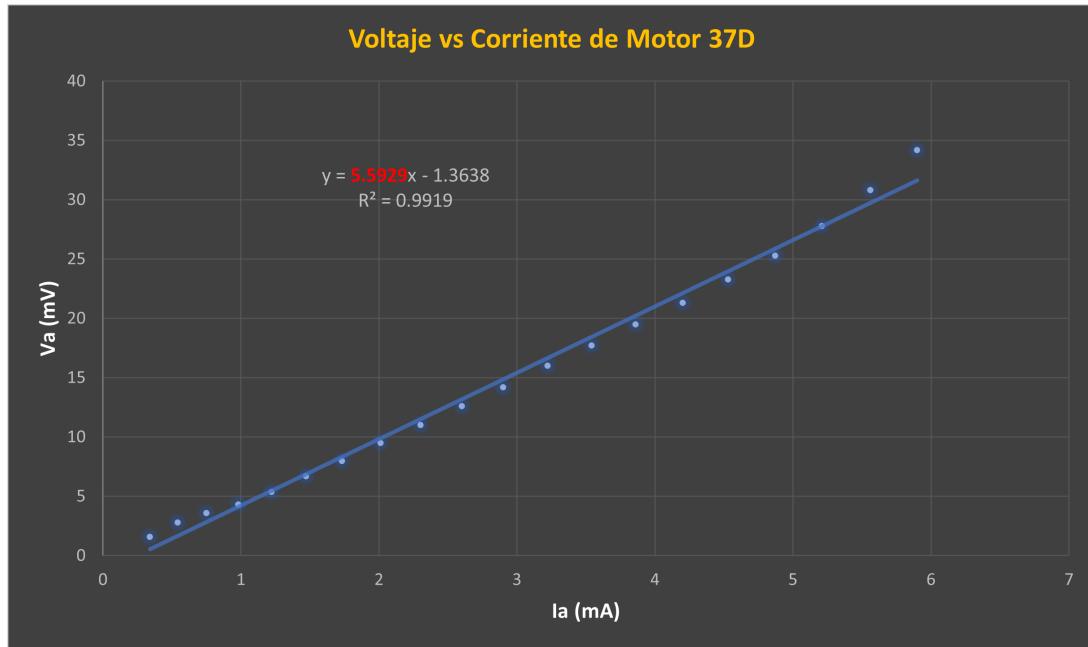


Figura 2.19: Verificación de la linealidad de la relación de voltaje y corriente

- Inductancia de armadura: $V_2 = 37\%V_{max} = 0,37 \times 1,13 = 0,4181 V$

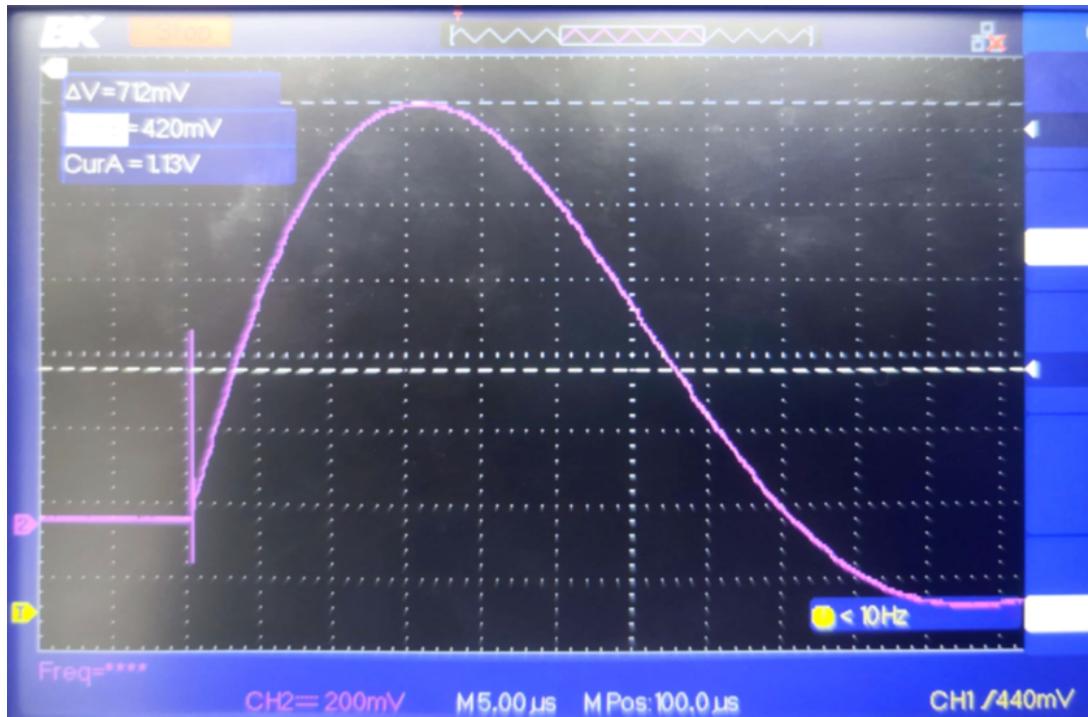


Figura 2.20: Identificación de voltajes entre los que se medirá el tiempo

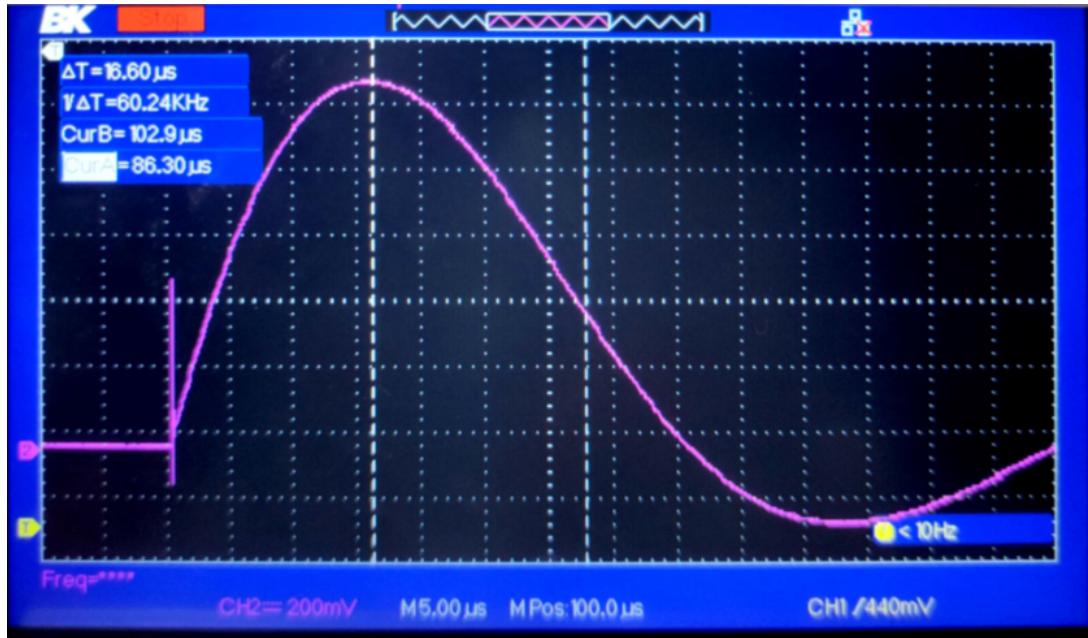


Figura 2.21: Medición de la constante de tiempo: $\tau = 16,6 \mu s$

La ecuación de constante de tiempo para un circuito RL permite calcular:

$$L_a = 5,499614571 \text{ mH}$$

- Estimación de parámetros mediante Simulink: $CPR = 70 \times 64 = 4480$

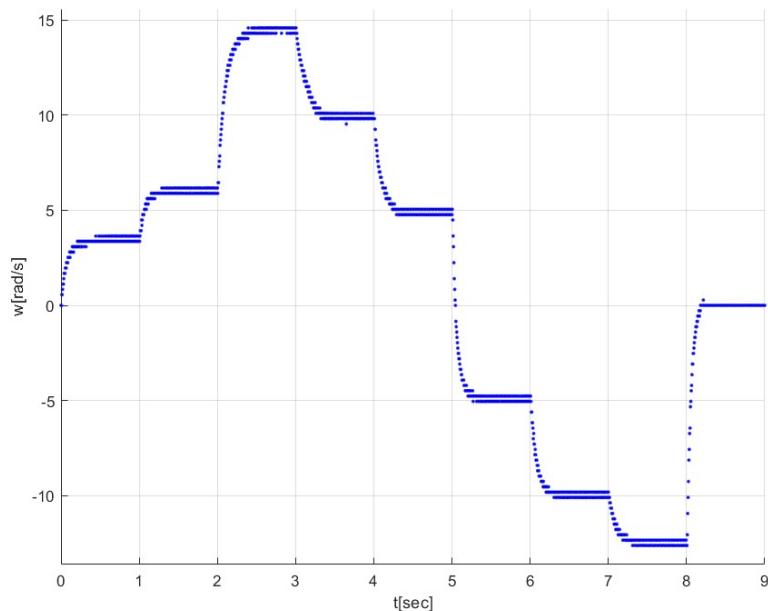


Figura 2.22: Data de velocidad angular obtenida mediante Arduino y Matlab

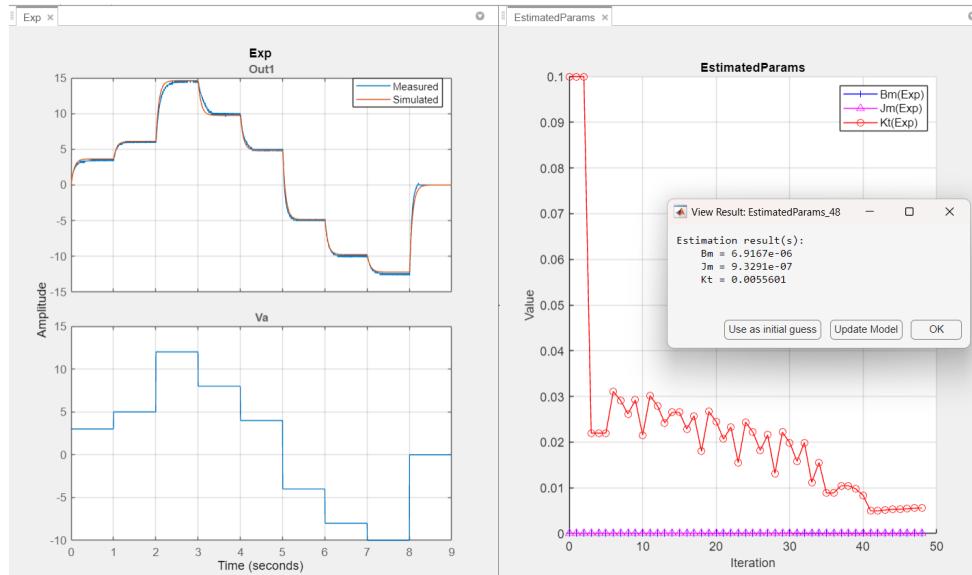


Figura 2.23: Parámetros que ajustan el modelo a la data de velocidad angular

$$B_m = 6,9167 \times 10^{-6}$$

$$J_m = 9,3291 \times 10^{-7}$$

$$K_t = K_b = 0,0055601$$

- Validación de los parámetros mediante el modelo en espacio estado continuo:

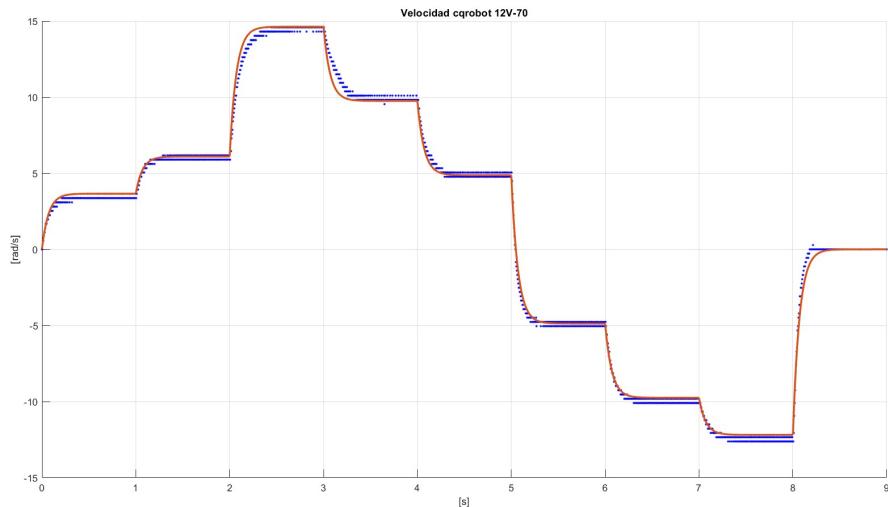


Figura 2.24: Validación de parámetros comparando modelo en espacio estado y la data

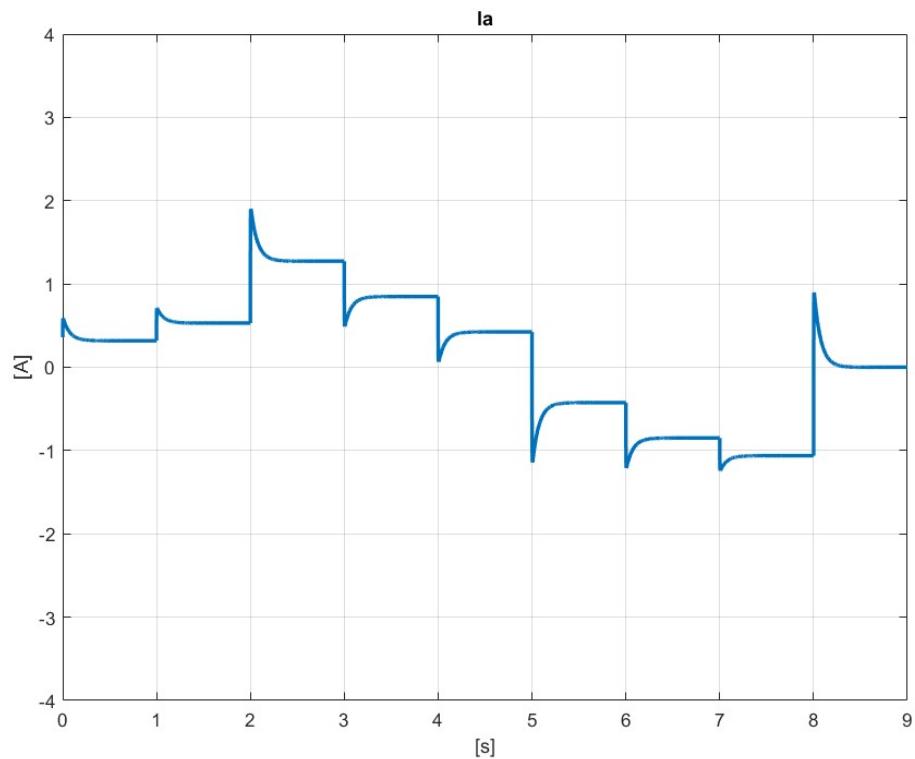


Figura 2.25: Corriente de armadura en respuesta al patrón de voltaje aplicado

2.2. Ecuaciones de cinemática directa

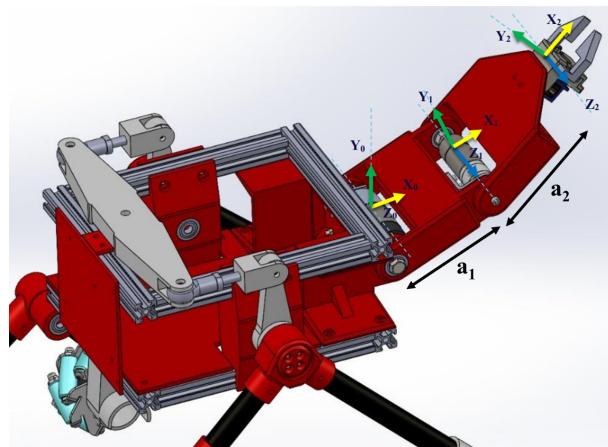


Figura 2.26: Sistemas Coordenados según DH-WP aplicados al manipulador

link	θ_i	d_i	a_i	α_i
1	θ_1	0	a_1	0
2	θ_2	0	a_2	0

Cuadro 1: Parámetros geométricos según Denavit-Hartenberg

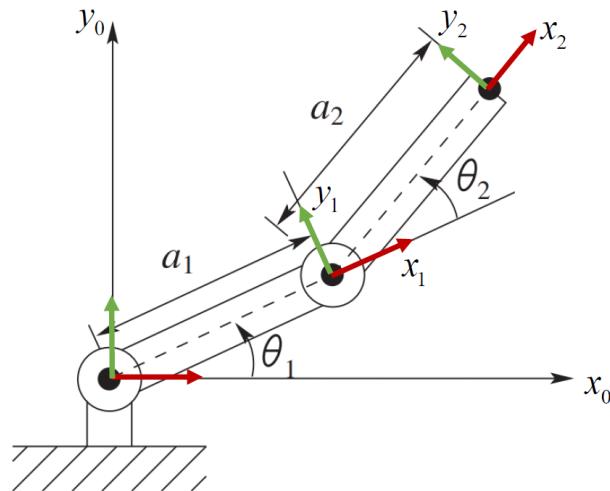


Figura 2.27: Modelo simple del manipulador planar de 2GDL

Notar que $\theta_1 = q_1$ y $\theta_2 = q_2$. Con ello en mente, se procede a calcular las matrices de transformación:

$$T_1^0 = A_1 = \begin{bmatrix} c_1 & -s_1 & 0 & a_1 c_1 \\ s_1 & c_1 & 0 & a_1 s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} c_2 & -s_2 & 0 & a_2 c_2 \\ s_2 & c_2 & 0 & a_2 s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \rightarrow T_2^0 = A_1 A_2 = \begin{bmatrix} c_{12} & -s_{12} & 0 & a_1 c_1 + a_2 c_{12} \\ s_{12} & c_{12} & 0 & a_1 s_1 + a_2 s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

donde $c_1 = \cos(q_1)$, $s_1 = \sin(q_1)$, $c_2 = \cos(q_2)$, $s_2 = \sin(q_2)$, $c_{12} = \cos(q_1 + q_2)$ y $s_{12} = \sin(q_1 + q_2)$.

2.3. Centros de masa de eslabones y motores

- Centros de masa de los eslabones

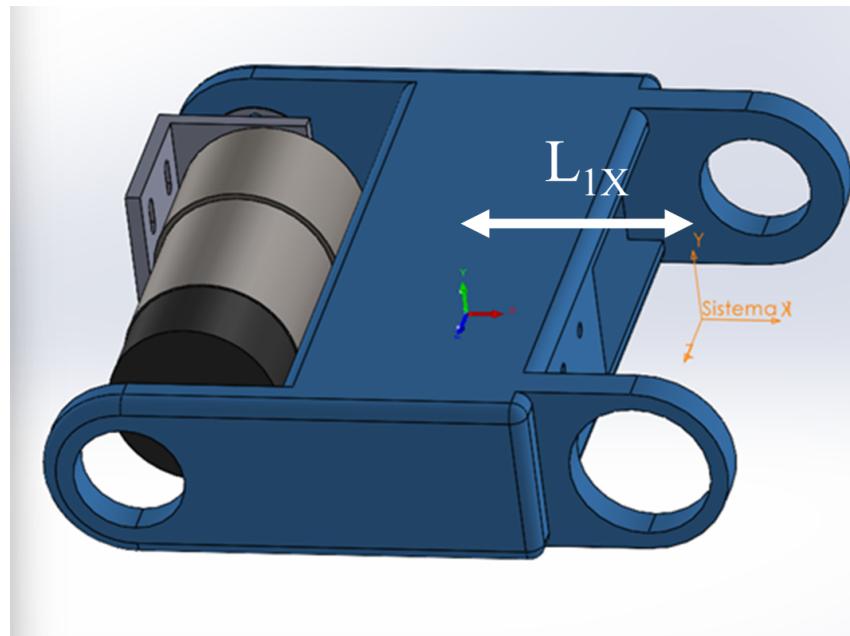


Figura 2.28: Distancia del CM de eslabón 1 respecto a sistema de coordenadas 1

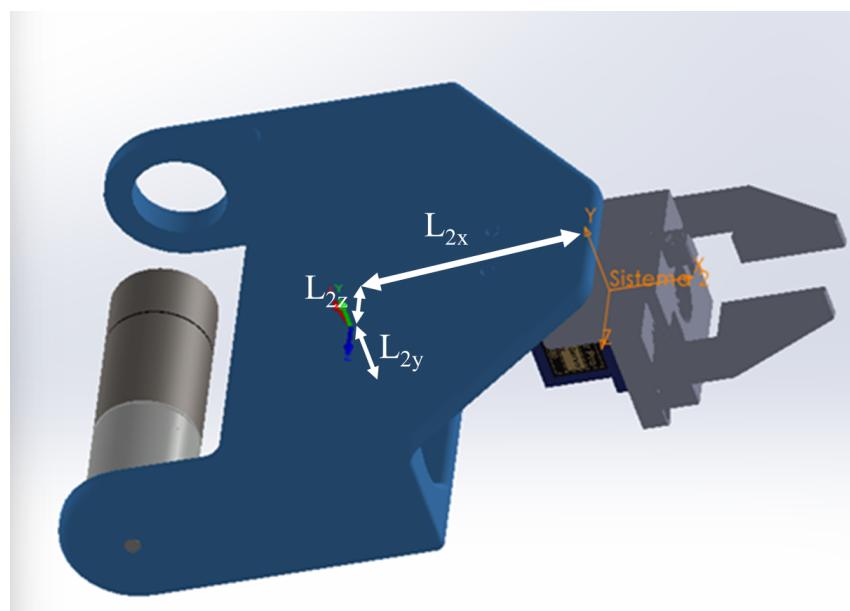


Figura 2.29: Distancias del CM de eslabón 2 respecto a sistema de coordenadas 2

Vectorialmente se obtienen las siguientes coordenadas:

$$p_{L_1}^1 = \begin{bmatrix} -L_{1x} \\ 0 \\ 0 \end{bmatrix} \quad p_{L_2}^2 = \begin{bmatrix} -L_{2x} \\ L_{2y} \\ L_{2z} \end{bmatrix}$$

Se aplica la matriz de transformación para obtener las coordenadas respecto a tierra.

$$p_{L_1}^0 = T_1^0 p_{L_1}^1 = \begin{bmatrix} (a_1 - L_{1x})c_1 \\ (a_1 - L_{1x})s_1 \\ 0 \end{bmatrix} \quad p_{L_2}^0 = T_2^0 p_{L_2}^2 = \begin{bmatrix} a_1c_1 + (a_2 - L_{2x})c_{12} - L_{2y}s_{12} \\ a_1s_1 + (a_2 - L_{2x})s_{12} + L_{2y}c_{12} \\ L_{2z} \end{bmatrix}$$

- Centros de masa de los motores

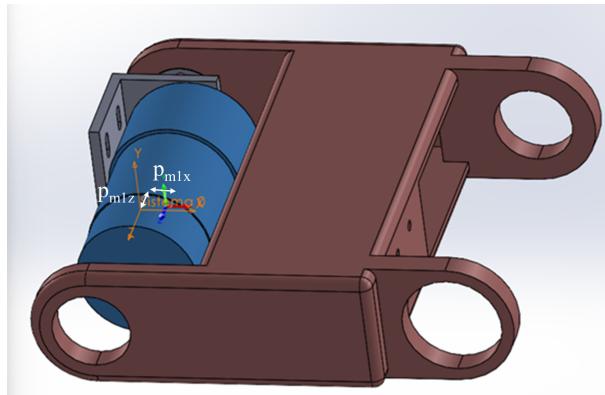


Figura 2.30: Distancia del CM de motor 1 respecto a sistema de coordenadas 0

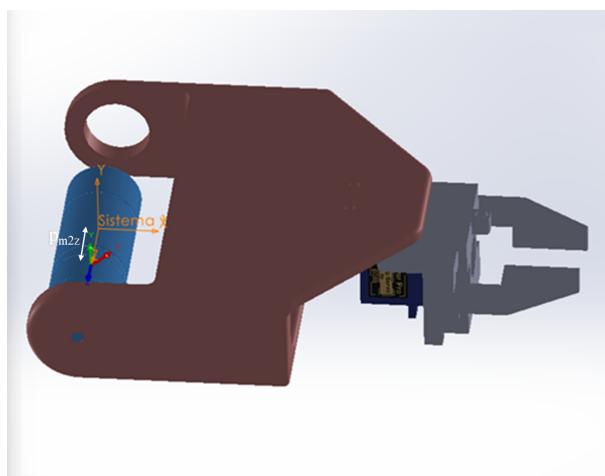


Figura 2.31: Distancias del CM de motor 2 respecto a sistema de coordenadas 1

Vectorialmente se obtienen las siguientes coordenadas:

$$p_{m_1}^0 = \begin{bmatrix} p_{m1x} \\ 0 \\ -p_{m1z} \end{bmatrix} \quad p_{m_2}^1 = \begin{bmatrix} 0 \\ 0 \\ p_{m2z} \end{bmatrix}$$

Se aplica la matriz de transformación para obtener las coordenadas respecto a tierra.

$$p_{m_1}^0 = \begin{bmatrix} p_{m1x} \\ 0 \\ -p_{m1z} \end{bmatrix} \quad p_{m_2}^0 = T_1^0 p_{m_2}^1 = \begin{bmatrix} a_1 c_1 \\ a_1 s_1 \\ p_{m2z} \end{bmatrix}$$

2.4. Jacobianos de eslabones y motores

- **Orígenes de coordenadas**

$$P_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad P_1 = \begin{bmatrix} a_1 c_1 \\ a_1 s_1 \\ 0 \end{bmatrix} \quad P_2 = \begin{bmatrix} a_1 c_1 + a_2 c_{12} \\ a_1 s_1 + a_2 s_{12} \\ 0 \end{bmatrix}$$

- **Vectores unitarios**

$$z_0 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad z_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad z_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

- **Vectores unitarios del eje de rotación de los motores**

$$z_{m1} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad z_{m2} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

- Jacobianos para los eslabones

$$J_v^{L_1} = \begin{bmatrix} z_0 \times (P_{L_1} - P_0) & 0 \end{bmatrix} \quad J_v^{L_1} = \begin{bmatrix} -(a_1 - L_{1x})s_1 & 0 \\ (a_1 - L_{1x})c_1 & 0 \\ 0 & 0 \end{bmatrix}$$

$$J_\omega^{L_1} = \begin{bmatrix} z_0 & 0 \end{bmatrix} \quad J_\omega^{L_1} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix}$$

$$J_v^{L_2} = \begin{bmatrix} z_0 \times (P_{L_2} - P_1) & z_1 \times (P_{L_2} - P_1) \end{bmatrix}$$

$$J_v^{L_2} = \begin{bmatrix} -a_1s_1 - L_{2y}c_{12} - (a_2 - L_{2x})s_{12} & -L_{2y}c_{12} - (a_2 - L_{2x})s_{12} \\ a_1c_1 - L_{2y}s_{12} + (a_2 - L_{2x})c_{12} & -L_{2y}s_{12} + (a_2 - L_{2x})c_{12} \\ 0 & 0 \end{bmatrix}$$

$$J_\omega^{L_2} = \begin{bmatrix} z_0 & z_1 \end{bmatrix} \quad J_\omega^{L_2} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}$$

- Jacobianos para los motores

$$J_v^{m_1} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$J_\omega^{m_1} = \begin{bmatrix} k_{r1}z_{m1} & 0 \end{bmatrix} \quad J_\omega^{m_1} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ k_{r1} & 0 \end{bmatrix}$$

$$J_v^{m_2} = \begin{bmatrix} z_0 \times (P_{m2} - P_0) & 0 \end{bmatrix} \quad J_v^{m_2} = \begin{bmatrix} -a_1s_1 & 0 \\ a_1c_1 & 0 \\ 0 & 0 \end{bmatrix}$$

$$J_\omega^{m_2} = \begin{bmatrix} z_0 & k_{r2}z_{m2} \end{bmatrix} \quad J_\omega^{m_2} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & k_{r2} \end{bmatrix}$$

2.5. Matrices de inercia de los eslabones y motores

- Matrices de Inercia de los eslabones - Componente translacional

$$D_v^{L_1} = m_{L_1} J_v^{L_1 T} J_v^{L_1} = \begin{bmatrix} m_{L_1} (L_{1x} - a_1)^2 & 0 \\ 0 & 0 \end{bmatrix}$$

$$D_v^{L_2} = m_{L_2} J_v^{L_2 T} J_v^{L_2} = \begin{bmatrix} d_1 & d_2 \\ d_3 & d_4 \end{bmatrix}$$

Donde:

$$d_1 = m_{L_2}((a_2 - L_{2x})^2 + (a_1 - L_{2y})^2 + 2(1 - s_2)L_{2y}a_1 + 2(a_2 - L_{2x})c_2a_1$$

$$d_2 = m_{L_2}((a_2 - L_{2x})^2 + (a_2 - L_{2x})a_1c_2 + L_{2y}^2 - a_1s_1L_{2y})$$

$$d_3 = m_{L_2}((a_2 - L_{2x})^2 + L_{2y}^2 + (a_2 - L_{2x})a_1c_2 - a_1s_2L_{2y})$$

$$d_4 = m_{L_2}((a_2 - L_{2x})^2 + L_{2y}^2)$$

- Matrices de Inercia de los motores - Componente translacional

$$D_v^{m_1} = m_{m_1} J_v^{m_1 T} J_v^{m_1} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$D_v^{m_2} = m_{m_2} J_v^{m_2 T} J_v^{m_2} = \begin{bmatrix} a_1^2 m_{m_2} & 0 \\ 0 & 0 \end{bmatrix}$$

- Matrices de Inercia de los eslabones - Componente rotacional

$$I_{L_1} = \begin{bmatrix} I_{1xx} & 0 & 0 \\ 0 & I_{1yy} & 0 \\ 0 & 0 & I_{1zz} \end{bmatrix}$$

$$D_\omega^{L_1} = J_\omega^{L_1 T} (R_1 I_{L_1} R_1^T) J_\omega^{L_1} = \begin{bmatrix} I_{1zz} & 0 \\ 0 & 0 \end{bmatrix}$$

$$I_{L_2} = \begin{bmatrix} I_{2xx} & 0 & 0 \\ 0 & I_{2yy} & 0 \\ 0 & 0 & I_{2zz} \end{bmatrix}$$

$$D_{\omega}^{L_2} = J_{\omega}^{L_2 T} (R_2 I_{L_2} R_2^T) J_{\omega}^{L_2} = \begin{bmatrix} I_{2zz} & I_{2zz} \\ I_{2zz} & I_{2zz} \end{bmatrix}$$

- Matrices de Inercia de los motores - Componente rotacional

$$I_{m_1} = \begin{bmatrix} I_{m1x} & 0 & 0 \\ 0 & I_{m1y} & 0 \\ 0 & 0 & I_{m1z} \end{bmatrix}$$

$$D_{\omega}^{m_1} = J_{\omega}^{m_1 T} (R_{m1} I_{m_1} R_{m1}^T) J_{\omega}^{m_1} = \begin{bmatrix} I_{m1z} k_{r1}^2 & 0 \\ 0 & 0 \end{bmatrix}$$

$$I_{m_2} = \begin{bmatrix} I_{m2x} & 0 & 0 \\ 0 & I_{m2y} & 0 \\ 0 & 0 & I_{m2z} \end{bmatrix}$$

$$D_{\omega}^{m_2} = J_{\omega}^{m_2 T} (R_{m2} I_{m_2} R_{m2}^T) J_{\omega}^{m_2} = \begin{bmatrix} I_{m2z} & I_{m2z} k_{r2} \\ I_{m2z} k_{r2} & I_{m2z} k_{r2}^2 \end{bmatrix}$$

2.6. Modelo dinámico

- Matriz D

$$D = D_v^{L_1} + D_v^{L_2} + D_{\omega}^{L_1} + D_{\omega}^{L_2} + D_v^{m_1} + D_v^{m_2} + D_{\omega}^{m_1} + D_{\omega}^{m_2} = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix}$$

Donde:

$$D_{11} = I_{1zz} + I_{2zz} + I_{m2z} + I_{m1z} k_{r1}^2 + a_1^2 m_{m2} + m_{L_1} (a_1 - L_{1x})^2 + m_{L_2} ((a_2 - L_{2x})^2 + L_{2y}^2 + a_1^2 - 2a_1(L_{2y}s_2 + (L_{2x} - a_2)c_2))$$

$$D_{12} = I_{2zz} + I_{m2z} k_{r2} + m_{L_2} ((a_2 - L_{2x})^2 + (a_2 - L_{2x})a_1 c_2 + L_{2y}^2 - a_1 s_2 L_{2y})$$

$$D_{21} = I_{2zz} + I_{m2z} k_{r2} + m_{L_2} ((a_2 - L_{2x})^2 + (a_2 - L_{2x})a_1 c_2 + L_{2y}^2 - a_1 s_2 L_{2y})$$

$$D_{22} = I_{2zz} + I_{m2z} k_{r2}^2 + m_{L_2} ((a_2 - L_{2x})^2 + L_{2y}^2)$$

■ Términos de Christoffel

$$c_{111} = 0$$

$$c_{112} = -a_1 m_{L_2} (L_{2y} c_2 + (a_2 - L_{2x}) s_2)$$

$$c_{121} = -a_1 m_{L_2} (L_{2y} c_2 + (a_2 - L_{2x}) s_2)$$

$$c_{122} = -a_1 m_{L_2} (L_{2y} c_2 + (a_2 - L_{2x}) s_2)$$

$$c_{211} = a_1 m_{L_2} (L_{2y} c_2 + (a_2 - L_{2x}) s_2)$$

$$c_{212} = 0$$

$$c_{221} = 0$$

$$c_{222} = 0$$

■ Matriz de Coriolis

$$C = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}$$

Donde:

$$C_{11} = -a_1 m_{L_2} (L_{2y} c_2 + (a_2 - L_{2x}) s_2) \dot{q}_2$$

$$C_{12} = -a_1 m_{L_2} (L_{2y} c_2 + (a_2 - L_{2x}) s_2) (\dot{q}_1 + \dot{q}_2)$$

$$C_{21} = a_1 m_{L_2} (L_{2y} c_2 + (a_2 - L_{2x}) s_2) \dot{q}_1$$

$$C_{22} = 0$$

■ Matriz G

$$g_0 = \begin{bmatrix} 0 \\ -g \\ 0 \end{bmatrix}$$

$$P_{L_1} = -m_{L_1} (g_0^T) p_{L_1} = g m_{L_1} s_1 (a_1 - L_{1x})$$

$$P_{L_2} = -m_{L_2} (g_0^T) p_{L_2} = g m_{L_2} (L_{2y} c_{12} + (a_2 - L_{2x}) s_{12} + a_1 s_1)$$

$$P_{m_1} = -m_{m_1} (g_0^T) p_{m_1} = 0$$

$$P_{m_2} = -m_{m_2} (g_0^T) p_{m_2} = a_1 g m_{m_2} s_1$$

$$U(q) = P_{L_1} + P_{L_2} + P_{m_1} + P_{m_2}$$

$$\frac{\partial U(q)}{\partial q_i} = g_i(q)$$

$$g = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} = \begin{bmatrix} a_1 g m_{m_2} c_1 - g m_{L_2} (c_{12}(L_{2x} - a_2) - a_1 c_1) - g m_{L_1} c_1 (L_{1x} - a_1) \\ g m_{L_2} c_{12} (a_2 - L_{2x}) \end{bmatrix}$$

■ Tabla de valores numéricos

Parámetro	Valor	Parámetro	Valor	Parámetro	Valor
a_1	101	L_{1x}	49,13	p_{m1x}	6,88
a_2	110	L_{2x}	62,62	p_{m1z}	2,21
k_{r1}	70	L_{2y}	2,02	p_{m2z}	9,04
k_{r2}	78,09	L_{2z}	0,97	m_{L_1}	118
I_{1xx}	242766,47	m_{L_2}	92	I_{2xx}	199117,65
I_{1yy}	587174,13	m_{m_1}	207	I_{2yy}	596783,5
I_{1zz}	380224,82	m_{m_2}	103	I_{2zz}	428323,99
I_{m1x}	107813,88	I_{m1y}	117773,71	I_{m1z}	42081,43
I_{m2x}	45103,51	I_{m2y}	45028,5	I_{m2z}	7895,83

Cuadro 2: Tabla de parámetros y valores

- Las distancias se encuentran en mm.
- Las masas se encuentran en g.
- las inercias se encuentrasn en g mm².

2.7. Simulación del modelo dinámico

Una vez hallado el modelo dinámico se procede a cargar las ecuaciones halladas así como los parámetros físicos necesarios del robot en Matlab para realizar las múltiples simulaciones. Entonces, se modifica los códigos siguientes que caracterizan el modelo dinámico del manipulador planar de 2GDL:

load_parameteres, robot_C, robot_D y robot_G

- **Dinámica directa.** Trayectoria de juntas en función del torque:

$$\tau_1 = 0,1, \tau_2 = 0,01, t \in [0, 2,5] \quad \tau_1 = -0,1, \tau_2 = -0,01, t \in [2,5, 5]$$

$$q_{1o} = -\pi/2, q_{2o} = 0 \quad \dot{q}_{1o} = 0, \dot{q}_{2o} = 0$$

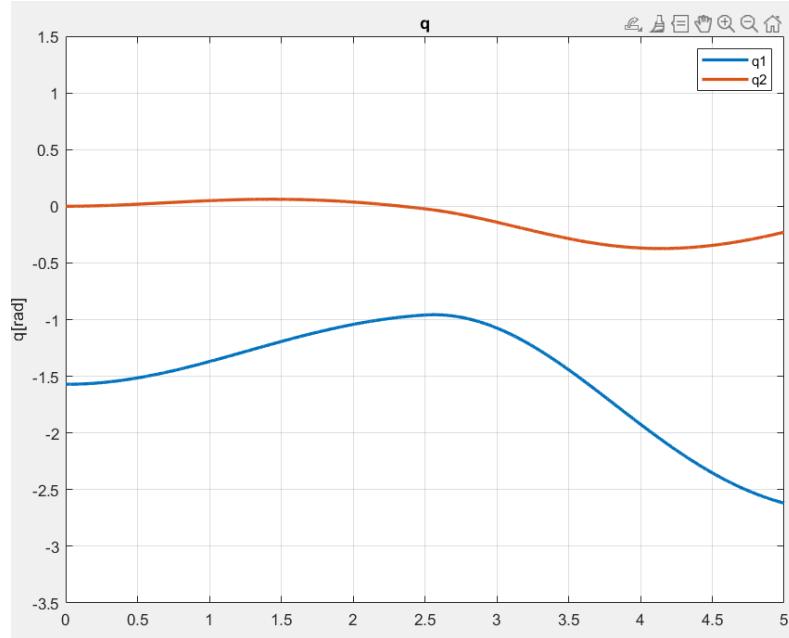


Figura 2.32: Posición de las juntas

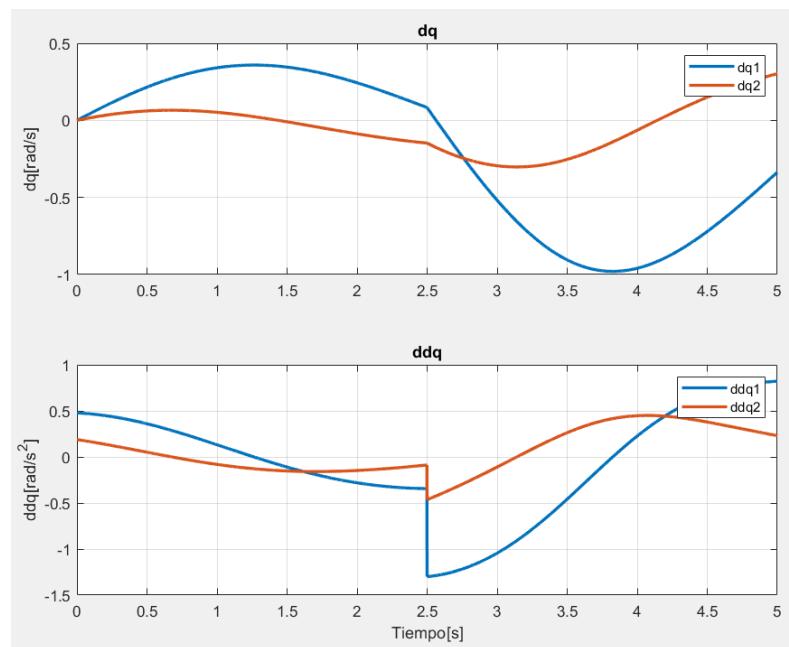


Figura 2.33: Velocidad y aceleración de las juntas

- **Dinámica inversa** Torques para perfil de velocidad trapezoidal:

$$t_f = 0,5 \quad \dot{q}_c = 4$$

$$q_{1o} = -\pi/2, q_{1f} = 0 \quad q_{2o} = 0, q_{2f} = \pi/2$$

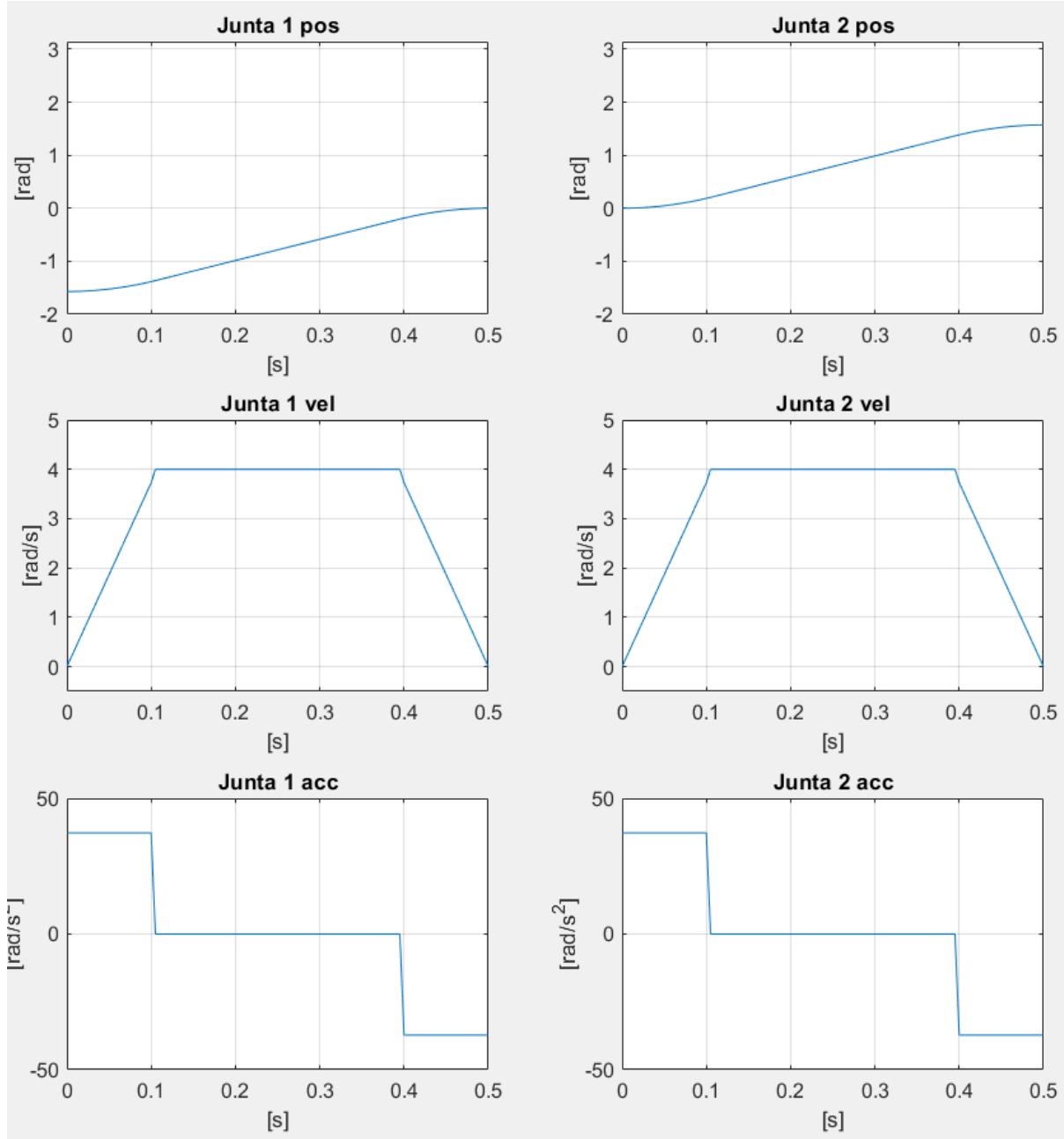


Figura 2.34: Posición, velocidad y aceleración de las juntas

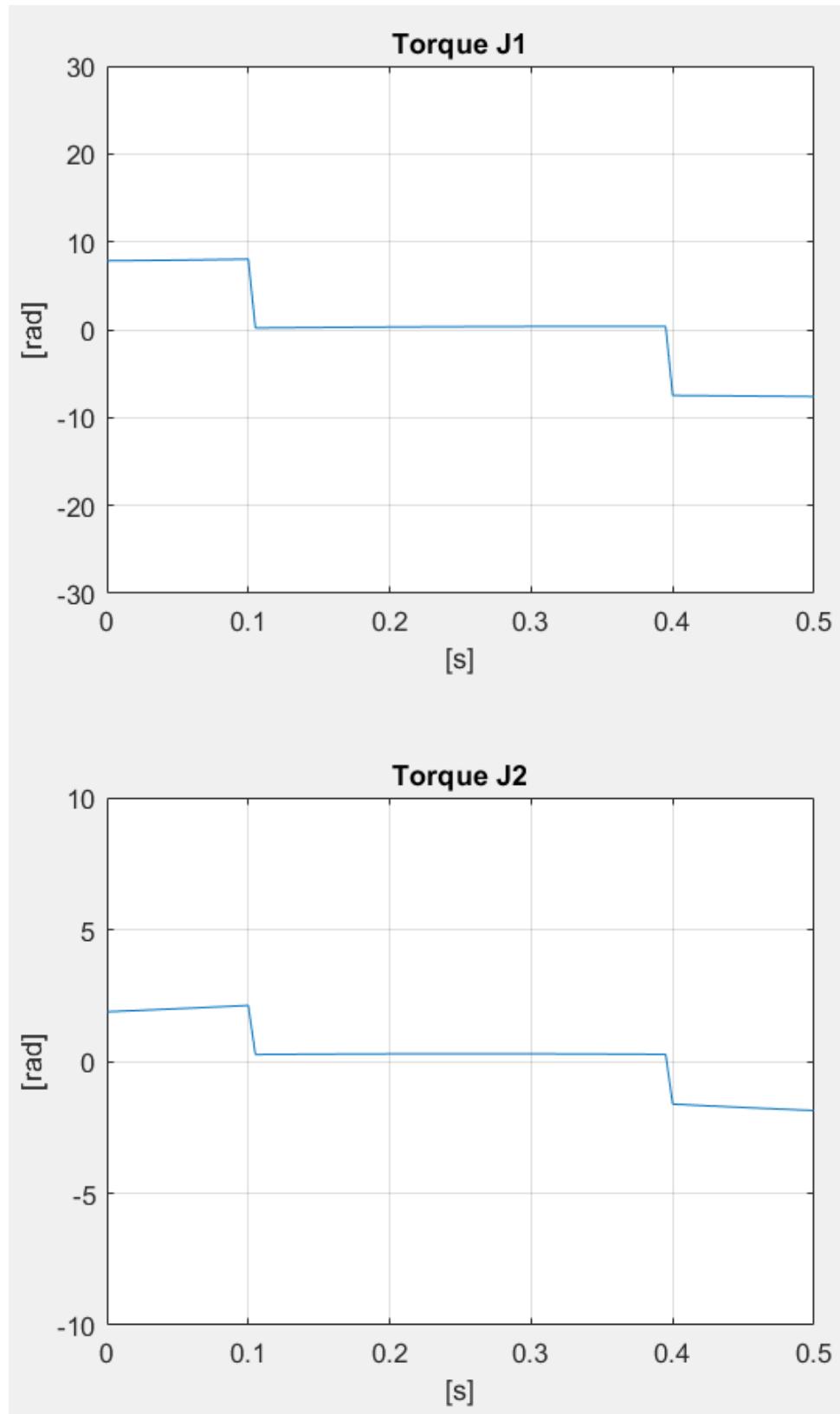


Figura 2.35: Torques de las juntas

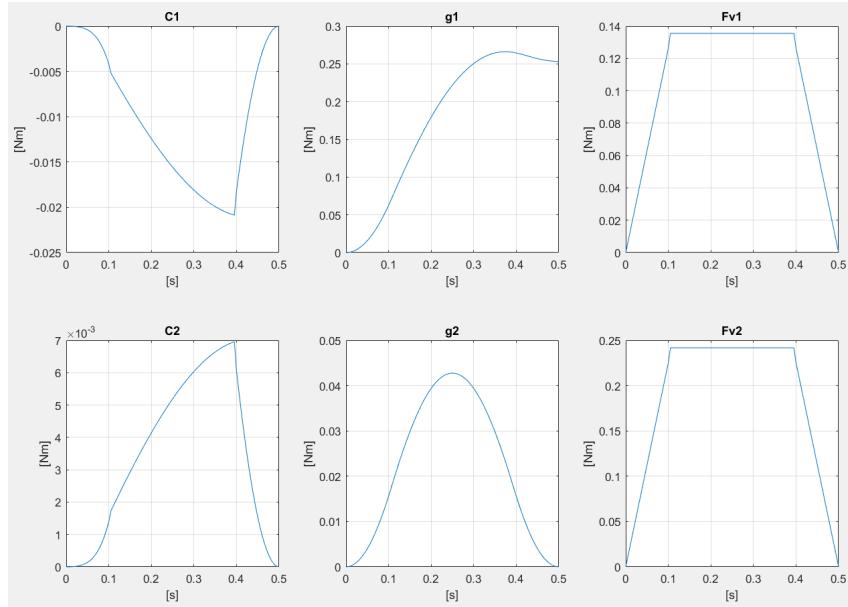


Figura 2.36: Contribución de Coriolis, gravedad y fricción

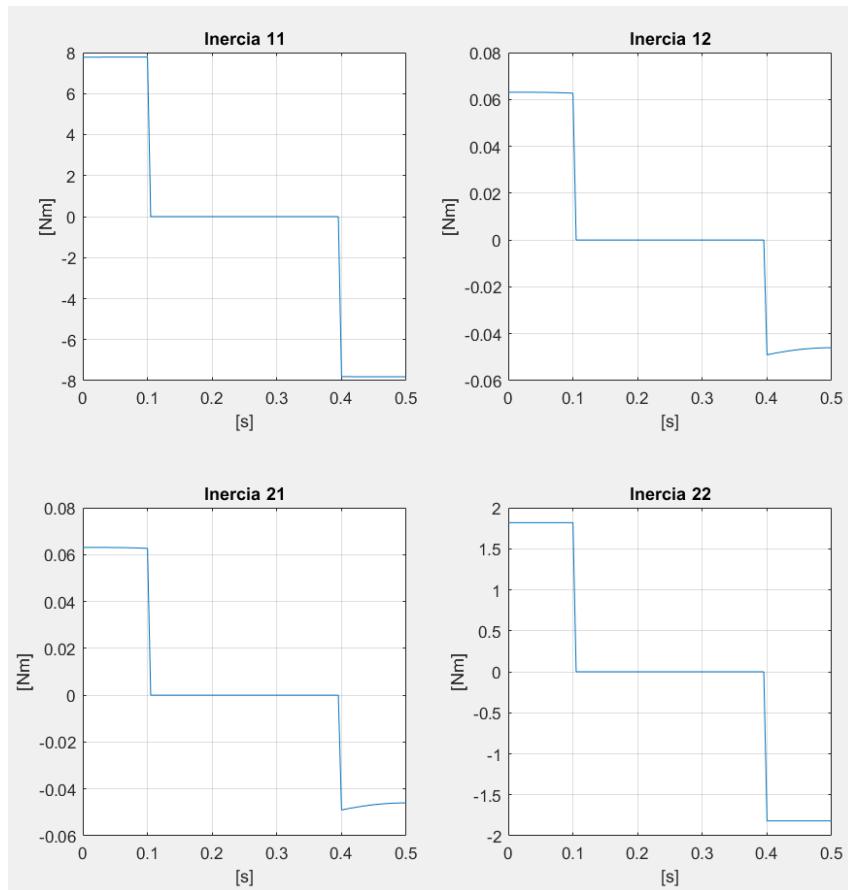


Figura 2.37: Matriz de inercia

- Sistema de actuación como sistema controlado por voltaje:

$$v_{c1} = 1,0, v_{c2} = 0,2, t \in [0, 2,5] \quad v_{c1} = 0,5, v_{c2} = 0,1, t \in [2,5, 5]$$

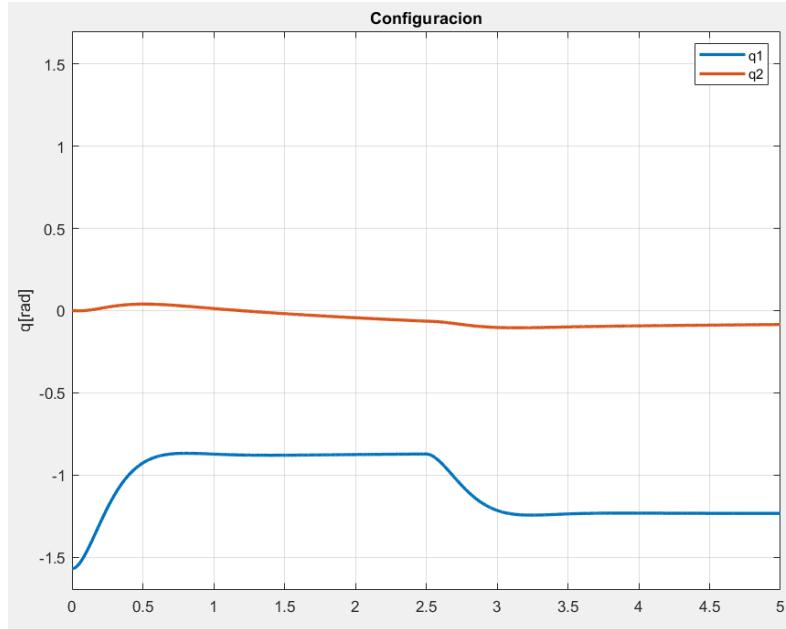


Figura 2.38: Posición de las juntas

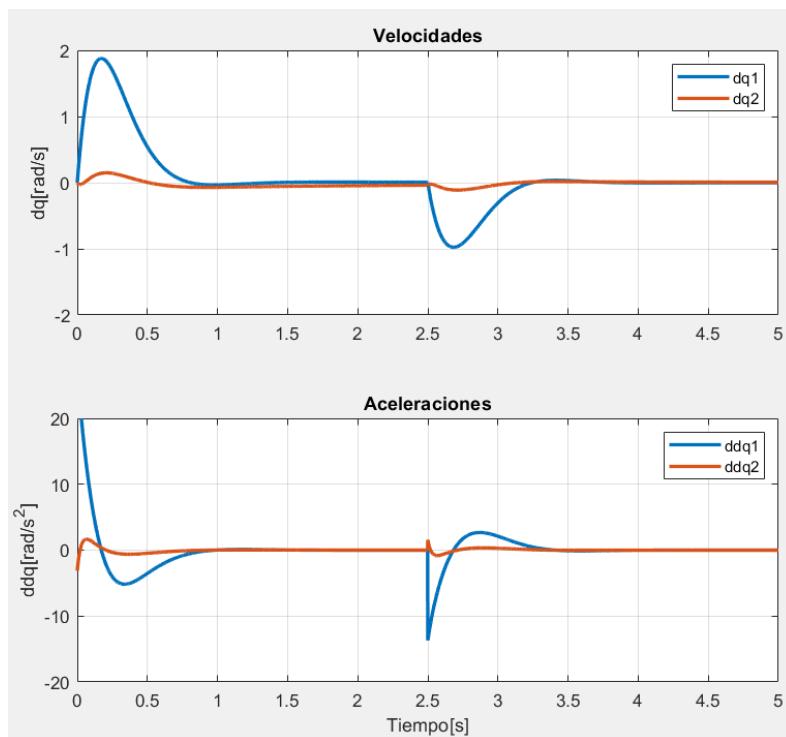


Figura 2.39: Velocidad y aceleración de las juntas

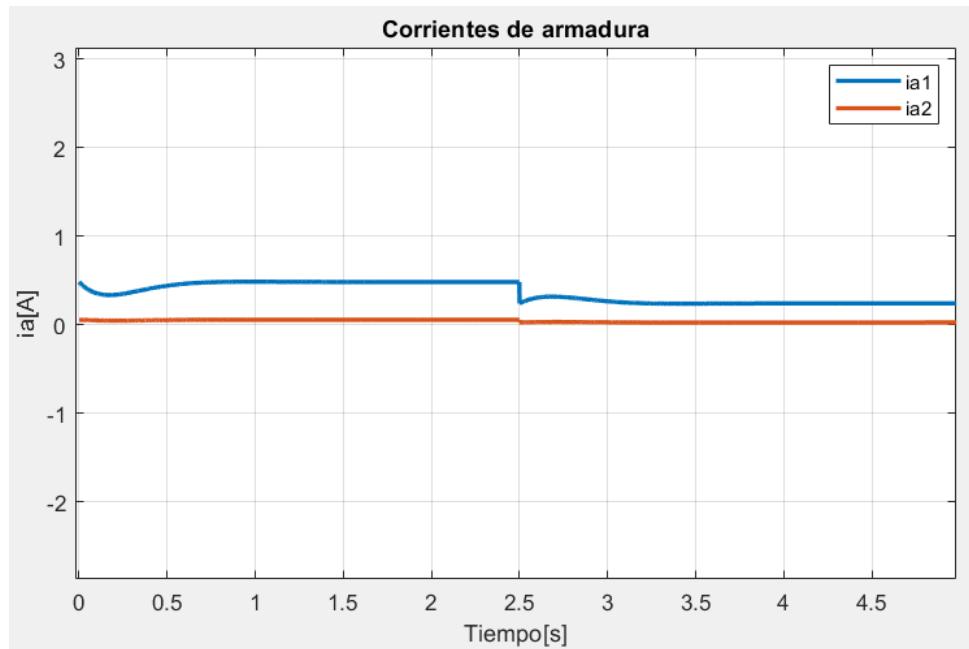


Figura 2.40: Corrientes de armadura en los motores

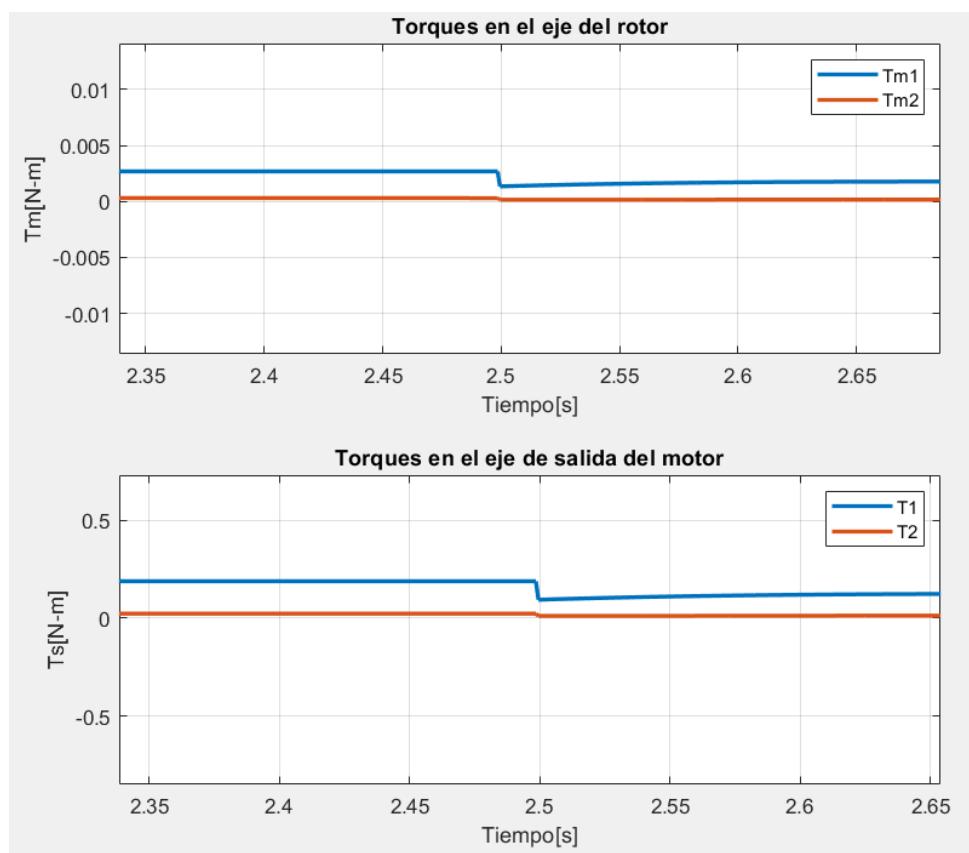


Figura 2.41: Torques en los motores

2.8. Validación de los motores

■ Requerimientos y Parámetros Motor 25D

Parámetro	Requerido	Valor máximo del motor	Cumple
Torque	0.3 N.m	0.5011 N.m	Sí
Velocidad	34 RPM	77 RPM	Sí
Corriente	0.80 A	1.89 A	Sí

■ Requerimientos y Parámetros Motor 37D

Parámetro	Requerido	Valor máximo del motor	Cumple
Torque	0.85 N.m	2.6477 N.m	Sí
Velocidad	34 RPM	150 RPM	Sí
Corriente	0.95 A	3 A	Sí

3. Control descentralizado

Video de demostración: Enlace

3.1. Análisis de resultados para diferentes configuraciones

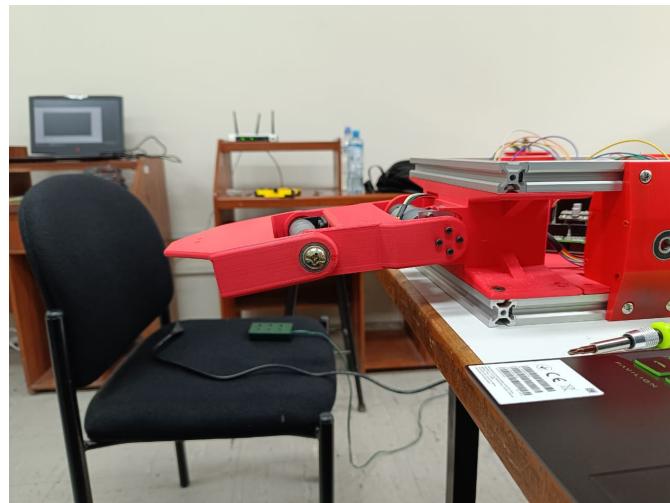


Figura 3.1: Configuración: $q_1 = 0$, $q_2 = 0$



Figura 3.2: Configuración: $q_1 = \pi/4$, $q_2 = 0$



Figura 3.3: Configuración: $q_1 = \pi/4$, $q_2 = \pi/6$



Figura 3.4: Configuración: $q_1 = -\pi/4$, $q_2 = 0$

3.2. Código ingresar variables deseadas en el espacio X

```

1 a1 = 101; % longitud del primer enlace
2 a2 = 110; % longitud del segundo enlace
3 % Leer las coordenadas del punto objetivo
4 x = input('Introduce la coordenada x del punto objetivo: ');
5 y = input('Introduce la coordenada y del punto objetivo: ');
6 % Calcular la distancia al punto objetivo
7 r = sqrt(x^2 + y^2);
8 % Verificar si el punto objetivo es alcanzable
9 if r > a1 + a2
10    disp('El punto objetivo no es alcanzable.');
11 else
12    % Calcular los ángulos de las articulaciones usando la
13    % cinemática inversa
14    c2 = (x^2 + y^2 - a1^2 - a2^2) / (2 * a1 * a2);
15    s2 = sqrt(1-c2^2);
16    theta2 = atan2(s2,c2);
17    theta1 = atan2(y, x) - atan2(a2 * s2, a1 + a2 * c2);
18    % Convertir los ángulos a grados
19    theta1 = rad2deg(theta1);
20    theta2 = rad2deg(theta2);
21
22    if abs(theta1) < 70 && abs(theta2) < 80
23        % Mostrar los ángulos de las articulaciones
24        fprintf('El ángulo de la primera articulación es %.2f
25                grados.\n', theta1);
26        fprintf('El ángulo de la segunda articulación es %.2f
27                grados.\n', theta2);
28    else
29        fprintf('Esa posición excede los límites mecánicos');
30    end
31 end

```

Bibliografía

- [1] Spong, M. W., Hutchinson, S., Vidyasagar, M. (2020). Robot Modeling and Control. John Wiley & Sons
- [2] Siciliano, B., Sciavicco, L., Villani, L., Oriolo, G. (2008). Robotics: Modelling, Planning and Control. Springer.

4. Anexos

4.1. Mediciones de masa



Figura 4.1: Masa del eslabón 1



Figura 4.2: Masa del eslabón 2



Figura 4.3: Masa del motor 1



Figura 4.4: Masa del motor 2