Addis Ababa Science and Technology University

Department of Electromechanical Engineering

Introduction to Robotics (EMEg5107)

Group Project Report

VEX V5 H-Drive Mobile Robot

Submitted by:

1. Sem Wogayehu	ETS 1462/14
2. Sena Mekonin	ETS 1466/14
3. Sinishaw Nigatu	ETS 1481/14
4. Sisay Aziz	ETS 1485/14
5. Tsegab Karta	ETS 1572/14
6. Yabetsh Girma	ETS 1608/14

Submitted to:

Mr. Mulat Tigabu

Course Instructor

Submission Date: December 28, 2025

Addis Ababa, Ethiopia

Abstract

This project presents a comprehensive analysis of the VEX V5 H-Drive mobile robot, a hybrid locomotion platform developed within the VEX Robotics Competition ecosystem. The robot employs a unique drivetrain configuration featuring two central traction wheels for differential drive propulsion and four corner-mounted omnidirectional wheels for stability and reduced rotational friction.The study encompasses dynamics and kinematics modeling i.e forward and inverse kinematic modeling, Jacobian derivation for velocity analysis.

Contents

1 Introduction

1.1 Background and Motivation

The evolution of mobile robotics has expanded beyond industrial applications into educational and competitive domains, where design constraints and performance requirements create unique engineering challenges. The VEX Robotics Competition (VRC) represents a global platform where student teams design, build, and program robots to solve dynamic game challenges. Within this ecosystem, the H-Drive configuration has emerged as an innovative solution that combines the simplicity of differential drive with enhanced maneuverability through omnidirectional support wheels.

This project builds upon our previous experience with the FANUC LR Mate 200iC stationary manipulator, transitioning from fixed-base kinematics to mobile platform dynamics. The VEX V5 H-Drive robot provides an ideal testbed for studying hybrid locomotion systems, offering practical insights into wheel-ground interactions, power efficiency optimization, and control system design under real-world constraints.

1.2 Problem Definition

Traditional differential drive mobile robots face significant limitations in competition environments:

- High scrubbing friction during turns, leading to energy inefficiency and unpredictable motion
- Wheel slippage causing odometry errors and inaccurate position estimation
- Limited agility in tight spaces due to skid-steer turning mechanics

1.3 Application and Task Description

The VEX V5 H-Drive robot is designed for competition environments with specific operational requirements:

Primary Competition Tasks:

- **Object Manipulation**: Precisely moving game elements (cubes, rings, balls) to designated zones
- **Field Navigation**: Efficient path following in a 12'×12' field with static and dynamic obstacles
- **Alliance Coordination**: Collaborative task execution with partner robots
- **Defensive Positioning**: Maintaining strategic positions while resisting opponent pushes

1.4 Related Work

Related Work

The design of the VEX V5 H-Drive configuration is deeply rooted in the practical, community-driven engineering of the VEX Robotics Competition (VRC) ecosystem. While foundational principles of mobile robot kinematics and control are well-established in academic literature [1, 2], the specific optimization of hybrid drivetrains like the H-Drive has primarily evolved through crowd-sourced innovation within the VRC community. Extensive design discussions, build logs, and performance analyses are documented in online forums and team notebooks [3], representing a significant repository of applied knowledge. This practical development is supported by the official VEX hardware and programming frameworks [4, 5], which provide the standardized components and software interfaces necessary for implementation. Recent academic work has begun to formally analyze the performance of such competition-born designs [6], highlighting their effectiveness in constrained, dynamic environments and bridging the gap between empirical competition strategies and formal robotic modeling [7].

1.5 Contributions of This Project

- To develop kinematics modeling of the VEX V5
- TO develop dynamic modeling of thre VEX V5
- address H -Drive mobile robot

2 System Description

2.1 Overview of the VEX V5 H-Drive

The VEX V5 H-Drive is a versatile, modular mobile robotics platform designed for educational and competitive applications. It employs a hybrid drivetrain configuration, combining a differential drive core with omnidirectional support to achieve a balance of traction, maneuverability, and stability. This design is engineered for dynamic tasks in structured environments like the VEX Robotics Competition (VRC) arena, where rapid acceleration, precise turning, and resilience to interaction are paramount. Its integrated control system, extensive sensor ecosystem, and user-friendly programming environment make it an ideal platform for implementing and testing mobile robotics algorithms.

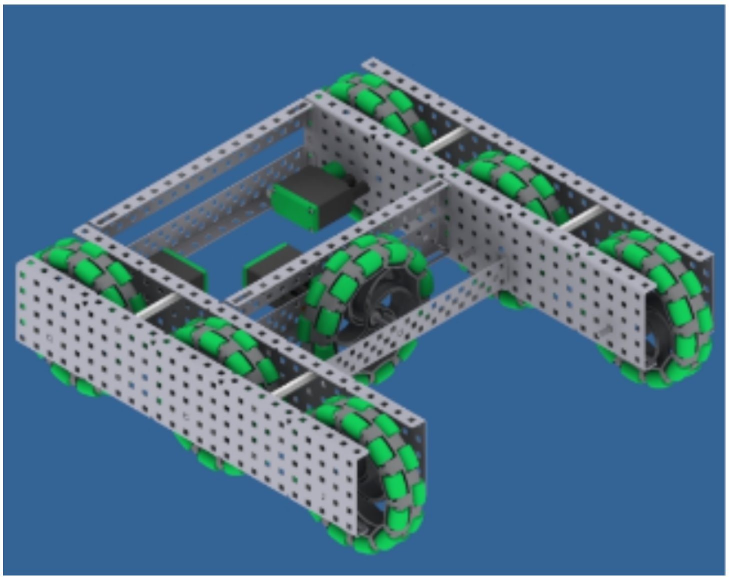

Figure 1: FANUC LR Mate 200iC Industrial Robot

2.2 Feature of the VEX V5 H-Drive

The platform has the following main characteristics:

- **Hybrid Drivetrain Configuration:** Features two central, motor-driven traction wheels for differential drive and four passive omnidirectional wheels at the corners for low-friction turning and support.
- **Modular Design:** Constructed from standardized VEX structural components, allowing for rapid reconfiguration and customization of the chassis and appendages.
- **Integrated Control System:** Centered on the V5 Robot Brain, which combines a processor, touchscreen display, and multiple smart ports for motors and sensors into a single unit.
- **High-Performance Actuation:** Utilizes V5 Smart Motors with integrated encoders, providing precise velocity and position control for accurate motion.
- **Comprehensive Software Suite:** Programmed using `VEXcode V5`, which supports block-based, Python, and C++ programming for all skill levels.

3 Methodology

3.1 Forward Kinematics

Forward kinematics determines the linear and angular velocity of the robot's center based on the rotational speeds of its two driven traction wheels.

For the two central traction wheels, the forward kinematic model is derived from the geometry of differential drive:

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ \frac{r}{L} & -\frac{r}{L} \end{bmatrix} \begin{bmatrix} \omega_r \\ \omega_l \end{bmatrix} \tag{1}$$

where:

- v: Linear velocity of the robot's center (m/s)
- ω: Angular velocity of the robot (rad/s)
- r: Radius of the traction wheels (m)
- L: Distance between the centers of the two traction wheels, known as the wheelbase (m)
- ω_r, ω_l: Angular velocities of the right and left traction wheels, respectively (rad/s)

The pose of the robot in the world coordinate frame $\{W\}$ is obtained by integrating these velocities:

$$\dot{x} = v \cos\theta \tag{2}$$

$$\dot{y} = v \sin\theta \tag{3}$$

$$\dot{\theta} = \omega \tag{4}$$

where (x, y) is the position of the robot's center and θ is its orientation.

3.2 Inverse Kinematics

Inverse kinematics calculates the required rotational speeds of each traction wheel to achieve a desired linear and angular velocity for the robot.

Solving Equation (1) for the wheel velocities yields:

$$\begin{bmatrix} \omega_r \\ \omega_l \end{bmatrix} = \begin{bmatrix} \frac{1}{r} & \frac{L}{2r} \\ \frac{1}{r} & -\frac{L}{2r} \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{5}$$

This equation is fundamental for robot control, as it translates a desired motion command (v, ω) into executable motor commands (ω_r, ω_l).

3.3 Parameter Specifications for VEX V5 H-Drive

The kinematic model uses the following physical parameters specific to a standard VEX V5 H-Drive configuration:

Parameter	**Value**	**Description**
Traction Wheel Radius (r)	0.0508 m	Radius of a standard 4-inch VEX wheel.
Wheelbase (L)	0.254 m	Typical distance between the centers of the two driven traction wheels (10 inches).
Maximum ω_r, ω_l	$\approx$ 31.4 rad/s	Based on a V5 Smart Motor's maximum free speed (200 RPM) under typical 36:60 gear reduction.
Corner Wheel Offset (d_x, d_y)	0.1905 m	Distance from the robot's center to each omnidirectional wheel along the X and Y axes (7.5 inches).
Omni Wheel Radius (r_{omni})	0.0413 m	Radius of a standard 3.25-inch VEX omni wheel.

Table 1: Kinematic parameters for the VEX V5 H-Drive model

3.4 Kinematic Constraints of Omnidirectional Wheels

The four omnidirectional wheels at the corners are passive and impose velocity constraints. For an omni wheel located at position $\mathbf{p}_i = [x_i, y_i]^T$ relative to the robot's center, its velocity $\mathbf{v}_{w,i}$ is given by:

$$\mathbf{v}_{w,i} = \mathbf{v} + \boldsymbol{\omega} \times \mathbf{r}_i \tag{6}$$

where $\mathbf{v} = [v_x, v_y]^T$ is the linear velocity vector of the robot's center, $\boldsymbol{\omega} = [0, 0, \omega]^T$ is the angular velocity vector, and $\mathbf{r}_i$ is the vector from the robot's center to the wheel. These constraints ensure the robot moves according to the differential drive model without slippage at the omni wheels.

To obtain the Jacobian matrix related to the linear velocity, it is necessary to compute the vector multiplication as described in Equation (6):

$$J_{V_n} = \ z_{n-1} \times (\ P_6^0 - \ P_{n-1}^0) \tag{7}$$

The Complete Jacobian Matrix is:

$$J = \begin{bmatrix} z_0 \times P_6^0 & z_1 \times (P_6^0 - P_1^0) & z_2 \times (P_6^0 - P_2^0) & z_3 \times (P_6^0 - P_3^0) & z_4 \times (P_6^0 - P_4^0) & z_5 \times (P_6^0 - P_5^0) \\ z_0 & z_1 & z_2 & z_3 & z_4 & z_5 \end{bmatrix}$$

3.5 Dynamic Model

Equations of Motion

The dynamic model of the VEX V5 H-Drive robot is derived using the Lagrangian formulation, considering the robot as a rigid body with two actuated wheels and four passive omni wheels. The generalized coordinates are defined as $\mathbf{q} = [x, y, \theta]^T$, where (x, y) is the position of the robot's center of mass (CoM) and θ is its orientation.

The Lagrangian $\mathcal{L}$ is defined as the difference between kinetic and potential energy:

$$\mathcal{L} = T - V \tag{8}$$

For motion on a horizontal plane, the potential energy V is constant and can be neglected. The kinetic energy T includes contributions from the translational motion of the chassis and the rotational motion of all wheels:

$$T = \frac{1}{2}m\mathbf{v}^T\mathbf{v} + \frac{1}{2}I\dot{\theta}^2 + \sum_{i=1}^{6}\left(\frac{1}{2}I_{w,i}\omega_{w,i}^2\right) \tag{9}$$

where:

- m: Total mass of the robot
- $\mathbf{v} = [\dot{x}, \dot{y}]^T$: Linear velocity vector of the CoM
- I: Moment of inertia of the robot about the vertical axis through the CoM
- $I_{w,i}$: Moment of inertia of wheel i about its axis of rotation
- $\omega_{w,i}$: Angular velocity of wheel i

Applying the Euler-Lagrange equation:

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial \mathcal{L}}{\partial \mathbf{q}} = \mathbf{Q} \tag{10}$$

where $\mathbf{Q}$ represents the generalized forces. This yields the equations of motion:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}(\dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) = \mathbf{B}\boldsymbol{\tau} \tag{11}$$

where:

- $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{3\times3}$: Inertia matrix (symmetric, positive definite)
- $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{3\times3}$: Coriolis and centrifugal matrix
- $\mathbf{F}(\dot{\mathbf{q}}) \in \mathbb{R}^3$: Friction forces vector
- $\mathbf{G}(\mathbf{q}) \in \mathbb{R}^3$: Gravitational forces vector (negligible on horizontal plane)
- $\mathbf{B} \in \mathbb{R}^{3\times2}$: Input transformation matrix
- $\boldsymbol{\tau} = [\tau_r, \tau_l]^T$: Torque vector applied to the right and left traction wheels

3.6 Inertia Matrix Derivation

The inertia matrix $\mathbf{M}(\mathbf{q})$ is derived from the kinetic energy expression. Considering the kinematic constraints of differential drive ($\dot{x} = v\cos\theta$, $\dot{y} = v\sin\theta$, $\dot{\theta} = \omega$), and expressing the wheel velocities in terms of robot velocities using Equation (1), the total kinetic energy can be written as:

$$T = \frac{1}{2}\begin{bmatrix} v & \omega \end{bmatrix} \mathbf{M}_r \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{12}$$

where $\mathbf{M}_r$ is the reduced inertia matrix in robot coordinates:

$$\mathbf{M}_r = \begin{bmatrix} m + \frac{2I_w}{r^2} & 0 \\ 0 & I + \frac{L^2}{2r^2} I_w \end{bmatrix} \tag{13}$$

Here, I_w is the inertia of each traction wheel (assumed identical), and we've neglected the small contribution of the omni wheels to the rotational inertia. The full inertia matrix in world coordinates is obtained through coordinate transformation:

$$\mathbf{M}(\mathbf{q}) = \mathbf{J}^T \mathbf{M}_r \mathbf{J} \tag{14}$$

where $\mathbf{J}$ is the Jacobian matrix that relates robot velocities to generalized velocities:

$$\mathbf{J} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \tag{15}$$

3.7 Coriolis and Centrifugal Matrix

The Coriolis and centrifugal matrix $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is derived using Christoffel symbols of the first kind [8]:

$$c_{jk} = \sum_{i=1}^{3} \Gamma_{ijk} \dot{q}_i, \quad \text{where} \quad \Gamma_{ijk} = \frac{1}{2}\left(\frac{\partial m_{kj}}{\partial q_i} + \frac{\partial m_{ki}}{\partial q_j} - \frac{\partial m_{ij}}{\partial q_k}\right) \tag{16}$$

For the VEX V5 H-Drive, since the inertia matrix $\mathbf{M}(\mathbf{q})$ depends only on the orientation θ, the Coriolis matrix simplifies to:

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} 0 & 0 & -m\dot{y} \\ 0 & 0 & m\dot{x} \\ 0 & 0 & 0 \end{bmatrix} \tag{17}$$

This matrix satisfies the skew-symmetry property $\dot{\mathbf{M}} - 2\mathbf{C}$ being skew-symmetric, which is crucial for control system stability proofs [9].

3.8 Friction Modeling

The friction force vector $\mathbf{F}(\dot{\mathbf{q}})$ includes viscous friction, Coulomb friction, and rolling resistance:

$$\mathbf{F}(\dot{\mathbf{q}}) = \mathbf{B}_v \dot{\mathbf{q}} + \mathbf{F}_c \cdot \text{sgn}(\dot{\mathbf{q}}) + \mathbf{F}_r \tag{18}$$

where:

- $\mathbf{B}_v \in \mathbb{R}^{3\times3}$: Viscous friction coefficient matrix (diagonal)
- $\mathbf{F}_c \in \mathbb{R}^3$: Coulomb friction coefficients
- $\mathbf{F}_r \in \mathbb{R}^3$: Rolling resistance coefficients
- $\text{sgn}(\cdot)$: Signum function

For the VEX V5 H-Drive on a VRC field surface (foam tiles), the dominant friction components are viscous damping at the wheel bearings and rolling resistance of the omni wheels [3].

3.9 Motor and Actuator Dynamics

Each V5 Smart Motor has its own dynamics. The torque τ_i produced by motor i is related to the input voltage V_i and current I_i by:

$$\tau_i = K_t I_i \tag{19}$$

$$V_i = RI_i + L\frac{dI_i}{dt} + K_e\omega_{m,i} \tag{20}$$

where:

- K_t: Motor torque constant (N·m/A)
- K_e: Back EMF constant (V·s/rad)
- R: Armature resistance ()
- L: Armature inductance (H)
- $\omega_{m,i}$: Angular velocity of motor i

The motor velocity $\omega_{m,i}$ is related to the wheel velocity $\omega_{w,i}$ through the gear ratio G:

$$\omega_{w,i} = G\omega_{m,i}, \quad \tau_{w,i} = \frac{\tau_{m,i}}{G} \tag{21}$$

Combining these with the robot dynamics yields the complete system model including actuator dynamics [10].

3.10 Dynamic Parameters for VEX V5 H-Drive

The following parameters are estimated based on VEX V5 components and typical competition robot configurations:

Parameter	**Value**	**Description**
Total mass (m)	5.0 kg	Includes chassis, motors, batteries, and manipulator
Chassis inertia (I)	0.15 kg·m^2	About vertical axis through CoM
Traction wheel inertia (I_w)	0.002 kg·m^2	About wheel axis
Viscous friction ($B_{v,xx}$, $B_{v,yy}$)	0.8 N·s/m	Linear motion damping
Viscous friction ($B_{v,\theta\theta}$)	0.05 N·m·s/rad	Rotational damping
Coulomb friction ($F_{c,x}$, $F_{c,y}$)	0.5 N	Linear motion static friction
Coulomb friction ($F_{c,\theta}$)	0.03 N·m	Rotational static friction
Motor torque constant (K_t)	0.018 N·m/A	V5 Smart Motor specification
Back EMF constant (K_e)	0.018 V·s/rad	V5 Smart Motor specification
Armature resistance (R)	0.6	V5 Smart Motor specification
Gear ratio (G)	36:60 (0.6)	Typical competition reduction

Table 2: Dynamic parameters for VEX V5 H-Drive model

3.11 Complete Dynamic Model

Combining all elements, the complete dynamic model of the VEX V5 H-Drive robot is:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}(\dot{\mathbf{q}}) = \mathbf{B}\boldsymbol{\tau} \tag{22}$$

$$\text{where } \boldsymbol{\tau} = \mathbf{G}\mathbf{K}_t\mathbf{I} \tag{23}$$

$$\mathbf{V} = \mathbf{R}\mathbf{I} + \mathbf{L}\frac{d\mathbf{I}}{dt} + \mathbf{K}_e\boldsymbol{\omega}_m \tag{24}$$

$$\boldsymbol{\omega}_w = \mathbf{G}\boldsymbol{\omega}_m \tag{25}$$

This model provides the foundation for designing advanced control strategies such as computed torque control, adaptive control, and robust control to handle uncertainties in parameters and disturbances during competition tasks.

3.12 Model Validation Approach

The dynamic model can be validated through:

1. **Simulation**: Comparing MATLAB/Simulink simulations with theoretical predictions
2. **Parameter Identification**: Using least-squares estimation from experimental data
3. **Control Performance**: Implementing model-based controllers and comparing with model-free approaches
4. **Energy Analysis**: Comparing predicted and actual battery consumption during maneuvers

3.13 Trajectory Generation

3.13.1 Point-to-Point Trajectory

For the 6 DOF FANUC LR Mate 200iC, quintic polynomial interpolation was selected based on the following technical requirements:

- Six Boundary Conditions: Each joint needs specification of:
  - Initial and final position
  - Initial and final velocity
  - Initial and final acceleration
- Continuity Requirements:
  - continuity (position, velocity, acceleration)
  - Prevents mechanical stress on servo motors
  - Maintains ±0.02 mm repeatability
- 6DOF-Specific Considerations:
  - All 6 joints move simultaneously
  - Synchronized motion requires smooth acceleration profiles
  - Prevents oscillation in Cartesian space
  - Respects individual joint speed limits (J6: 700°/s, J4: 450°/s, etc.)

The quintic polynomial provides the minimum degree needed to satisfy all 6 boundary conditions while ensuring smooth, predictable motion for all 6 degrees of freedom.

For each joint j (where $j = 1, 2, \dots, 6$), the trajectory is defined as:

$$q_j(t) = a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0 \tag{26}$$

The corresponding velocity and acceleration profiles are:

$$\dot{q}_j(t) = 5a_5 t^4 + 4a_4 t^3 + 3a_3 t^2 + 2a_2 t + a_1 \tag{27}$$

$$\ddot{q}_j(t) = 20a_5 t^3 + 12a_4 t^2 + 6a_3 t + 2a_2 \tag{28}$$

For Joint 1 of the FANUC LR Mate 200iC, consider the following specifications:

- Initial time: $t_0 = 1$ s
- Final time: $t_f = 3$ s
- Initial position: $q_0 = 10^\circ$ (≈ 0.1745 rad)
- Final position: $q_f = 60^\circ$ (≈ 1.0472 rad)
- Initial and final velocity: $\dot{q}_0 = \dot{q}_f = 0$
- Initial and final acceleration: $\ddot{q}_0 = \ddot{q}_f = 0$

Substituting the boundary conditions into equations (26), (27), and (28) gives the following system:

$$\begin{bmatrix} 1^5 & 1^4 & 1^3 & 1^2 & 1 & 1 \\ 3^5 & 3^4 & 3^3 & 3^2 & 3 & 1 \\ 5(1^4) & 4(1^3) & 3(1^2) & 2(1) & 1 & 0 \\ 5(3^4) & 4(3^3) & 3(3^2) & 2(3) & 1 & 0 \\ 20(1^3) & 12(1^2) & 6(1) & 2 & 0 & 0 \\ 20(3^3) & 12(3^2) & 6(3) & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_5 \\ a_4 \\ a_3 \\ a_2 \\ a_1 \\ a_0 \end{bmatrix} = \begin{bmatrix} 0.1745 \\ 1.0472 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Which simplifies to:

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 243 & 81 & 27 & 9 & 3 & 1 \\ 5 & 4 & 3 & 2 & 1 & 0 \\ 405 & 108 & 27 & 6 & 1 & 0 \\ 20 & 12 & 6 & 2 & 0 & 0 \\ 540 & 108 & 18 & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_5 \\ a_4 \\ a_3 \\ a_2 \\ a_1 \\ a_0 \end{bmatrix} = \begin{bmatrix} 0.1745 \\ 1.0472 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Solving this linear system yields the coefficients:

$$a_5 = -0.0022, \quad a_4 = 0.0273, \quad a_3 = -0.1091$$
$$a_2 = 0.1455, \quad a_1 = 0, \quad a_0 = 0.1128$$

Thus, the quintic polynomial trajectory for Joint 1 is:

$$q_1(t) = -0.0022t^5 + 0.0273t^4 - 0.1091t^3 + 0.1455t^2 + 0.1128 \tag{29}$$

The corresponding velocity and acceleration profiles are:

$$\dot{q}_1(t) = -0.0110t^4 + 0.1092t^3 - 0.3273t^2 + 0.2910t \tag{30}$$
$$\ddot{q}_1(t) = -0.0440t^3 + 0.3276t^2 - 0.6546t + 0.2910 \tag{31}$$

Position:

$$q(t) = a_5t^5 + a_4t^4 + a_3t^3 + a_2t^2 + a_1t + a_0 \tag{32}$$

Velocity:

$$\dot{q}(t) = 5a_5t^4 + 4a_4t^3 + 3a_3t^2 + 2a_2t + a_1 \tag{33}$$

Acceleration:

$$\ddot{q}(t) = 20a_5t^3 + 12a_4t^2 + 6a_3t + 2a_2 \tag{34}$$

System of Equations

$$q_0 = a_5t_0^5 + a_4t_0^4 + a_3t_0^3 + a_2t_0^2 + a_1t_0 + a_0$$

$$q_f = a_5t_f^5 + a_4t_f^4 + a_3t_f^3 + a_2t_f^2 + a_1t_f + a_0$$

$$\dot{q}_0 = 5a_5t_0^4 + 4a_4t_0^3 + 3a_3t_0^2 + 2a_2t_0 + a_1$$

$$\dot{q}_f = 5a_5t_f^4 + 4a_4t_f^3 + 3a_3t_f^2 + 2a_2t_f + a_1$$

$$\ddot{q}_0 = 20a_5t_0^3 + 12a_4t_0^2 + 6a_3t_0 + 2a_2$$

$$\ddot{q}_f = 20a_5t_f^3 + 12a_4t_f^2 + 6a_3t_f + 2a_2$$

$$\begin{bmatrix} t_0^5 & t_0^4 & t_0^3 & t_0^2 & t_0 & 1 \\ t_f^5 & t_f^4 & t_f^3 & t_f^2 & t_f & 1 \\ 5t_0^4 & 4t_0^3 & 3t_0^2 & 2t_0 & 1 & 0 \\ 5t_f^4 & 4t_f^3 & 3t_f^2 & 2t_f & 1 & 0 \\ 20t_0^3 & 12t_0^2 & 6t_0 & 2 & 0 & 0 \\ 20t_f^3 & 12t_f^2 & 6t_f & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_5 \\ a_4 \\ a_3 \\ a_2 \\ a_1 \\ a_0 \end{bmatrix} = \begin{bmatrix} q_0 \\ q_f \\ \dot{q}_0 \\ \dot{q}_f \\ \ddot{q}_0 \\ \ddot{q}_f \end{bmatrix}$$

3.13.2 Trajectory Planning and Generation

The reference trajectory for the FANUC LR Mate 200iC (quintic) polynomial interpolation in joint space. Each joint trajectory $q_j(t)$ is defined as:

$$q_j(t) = a_5t^5 + a_4t^4 + a_3t^3 + a_2t^2 + a_1t + a_0$$

where the coefficients a_i are computed to satisfy six boundary conditions: initial and final positions, zero initial and final velocities, and zero initial and final accelerations. This formulation ensures C continuity (position, velocity, and acceleration) of the motion. The trajectory transitions the robot from its configuration $q_0 = [0, \pi/4, -\pi/4, 0, \pi/4, 0]^T$ to a target pick-and-place configuration $q_f = [\pi/6, \pi/6, -\pi/6, \pi/12, \pi/6, \pi/18]^T$, with smooth acceleration profiles that respect the mechanical limits of each joint while preventing unnecessary stress on actuators.

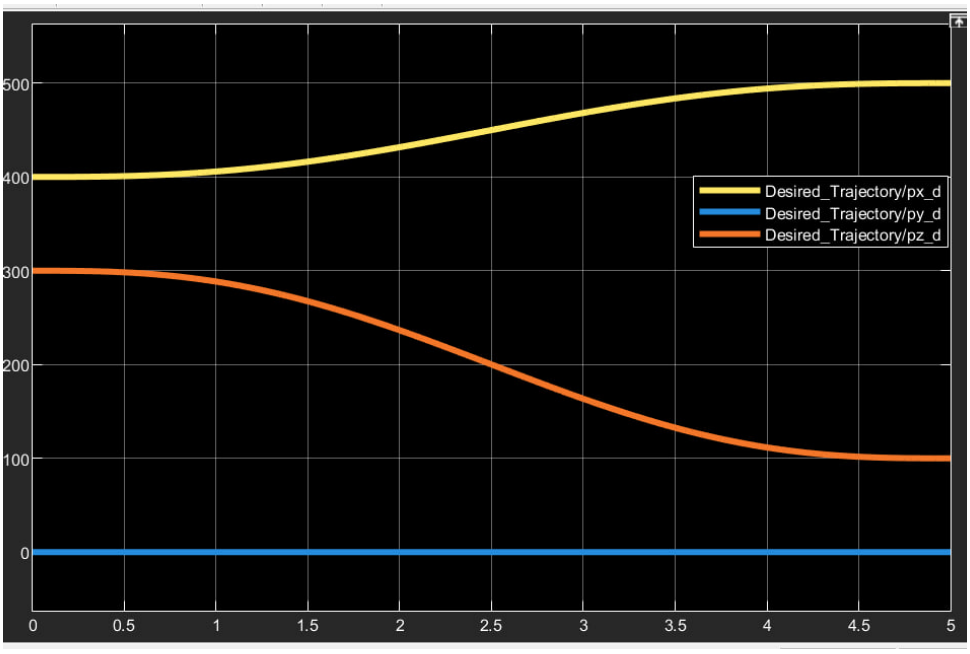

Figure 2: Desired joint trajectories

4 Control Design

4.1 Decentralized Control Method

Decentralized control is a joint-level strategy in which each actuator of a multi-DOF robot is governed independently using only its own local measurements. In the FANUC LR Mate 200iC, this approach simplifies controller design by treating each joint as an individual sub- system, while the mechanical structure and high gear ratios naturally reduce dynamic coupling among joints. Although nonlinear dynamics introduce some inter-axis interac- tions, decentralized PID control remains effective because each joint controller compensates for its own position, velocity, and integrated error, ensuring stable and accurate motion tracking with elimination of steady-state error. This method enhances modularity, supports high-frequency servo operation with integral anti-windup protection, and aligns with the control architecture commonly used in industrial robotic manipulators.

4.1.1 Simscape Multibody Model

The Simscape Multibody model represents the FANUC LR Mate 200iC using six revolute joints connected through rigid body links that follow the DH geometry. Joint torque inputs drive each revolute joint, while the model outputs joint angles, velocities, and torques. The World Frame defines the base, and transform sensing blocks extract the end-effector pose. This compact structure enables accurate kinematic and dynamic simulation of the robot inside MATLAB/Simulink for PID controller tuning and validation.

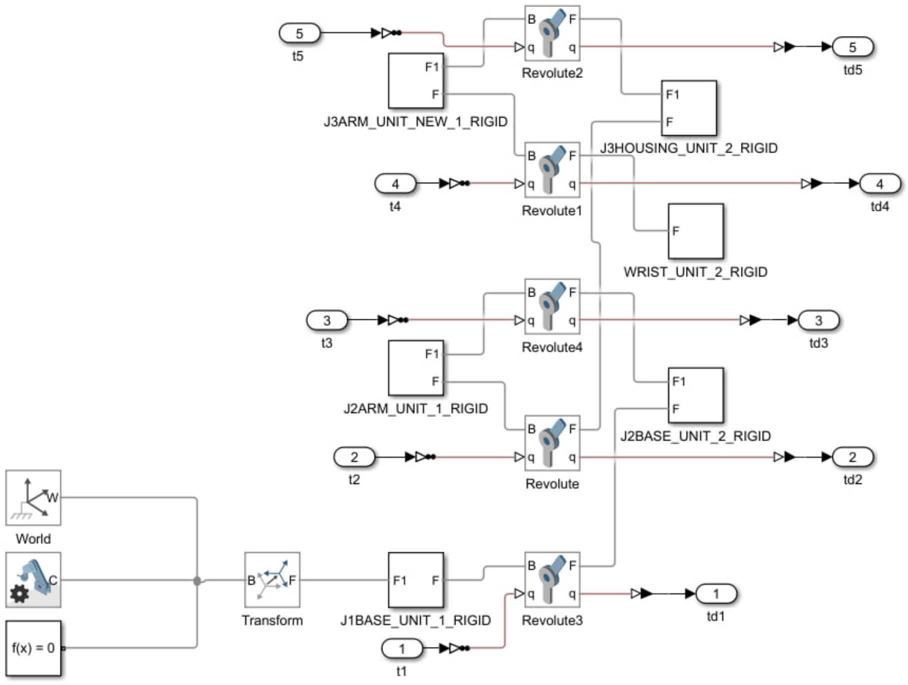

Figure 3: Simscape Multibody

4.1.2 Closed-Loop Control Structure

The Simulink control model integrates five key blocks: trajectory generation, inverse kinematics, PID controller, robot dynamic model, and forward kinematics verification. Trajectory generation produces Cartesian paths, converted to joint commands via inverse kinematics. The PID controller processes these references with actual joint states, computing torques with anti-windup. These drive the robot model while forward kinematics verifies end-effector accuracy against the original trajectory. Joint and Cartesian errors feed back through the gateway network, ensuring precise tracking with zero steady-state error across both motion spaces.

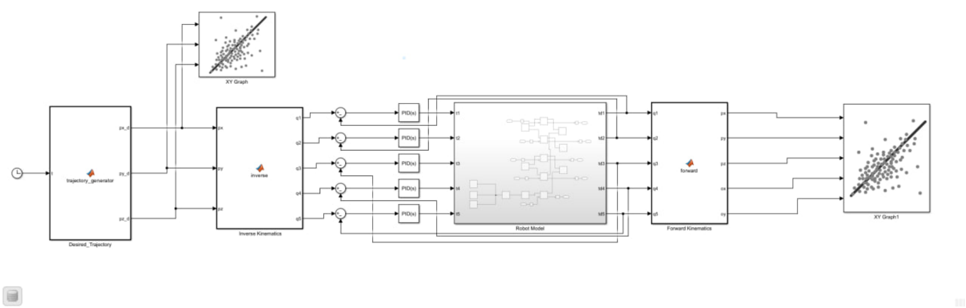

Figure 4: Closed loop Simulink

4.2 Centralized Control Method: Computed Torque Control

The Computed Torque Control method, is a centralized control approach that explicitly uses the robot's dynamic model to achieve precise trajectory tracking. Unlike decentralized PID control which treats each joint independently, Computed Torque Control accounts for the nonlinear coupling effects between joints, making it particularly suitable for high-speed, high-precision applications of the FANUC LR Mate 200iC.

The dynamics of the 6-DOF manipulator are governed by the Euler-Lagrange equations:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau \tag{35}$$

where:

- $q, \dot{q}, \ddot{q} \in \mathbb{R}^6$: Joint position, velocity, and acceleration vectors
- $\tau \in \mathbb{R}^6$: Joint torque vector
- $M(q) \in \mathbb{R}^{6\times6}$: Inertia matrix (symmetric, positive-definite)
- $C(q,\dot{q})\dot{q} \in \mathbb{R}^6$: Coriolis and centrifugal force vector
- $G(q) \in \mathbb{R}^6$: Gravity torque vector

The computed torque control law is given by:

$$\tau = M(q)[\ddot{q}_d + K_v(\dot{q}_d - \dot{q}) + K_p(q_d - q)] + C(q,\dot{q})\dot{q} + G(q) \tag{36}$$

where:

- $q_d, \dot{q}_d, \ddot{q}_d$: Desired joint position, velocity, and acceleration
- $K_p = \text{diag}(k_{p1}, \ldots, k_{p6})$: Positive definite diagonal position gain matrix
- $K_v = \text{diag}(k_{v1}, \ldots, k_{v6})$: Positive definite diagonal velocity gain matrix

Substituting the control law (36) into the dynamic equation (35) yields the linear error dynamics:

$$\ddot{e} + K_v\dot{e} + K_p e = 0 \tag{37}$$

where $e = q_d - q$ is the tracking error. By properly choosing K_p and K_v, the system becomes globally asymptotically stable with decoupled second-order dynamics for each joint.

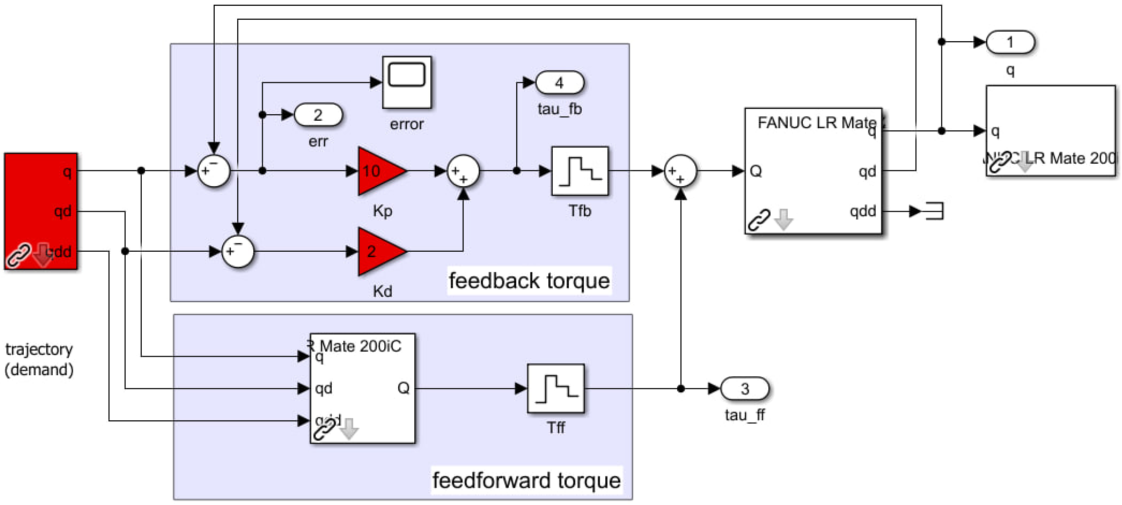

Figure 5: Computed Torque Control Simulink Implementation

5 Results and Discussion

The PID control simulation demonstrates effective trajectory tracking for all six joints of the FANUC LR Mate 200iC. Position errors converge to near-zero values ($< 0.05\,\text{rad}$) within 2-3 seconds and remain stable throughout the 5-second simulation, with smooth response characteristics across all joint trajectories. This satisfactory performance validates the decentralized PID controller architecture with independent joint tuning, providing reliable tracking for industrial applications. While slightly slower than computed torque control, the PID approach offers simpler implementation and reduced computational requirements, making it suitable for applications where moderate precision and robustness are prioritized over maximum speed.

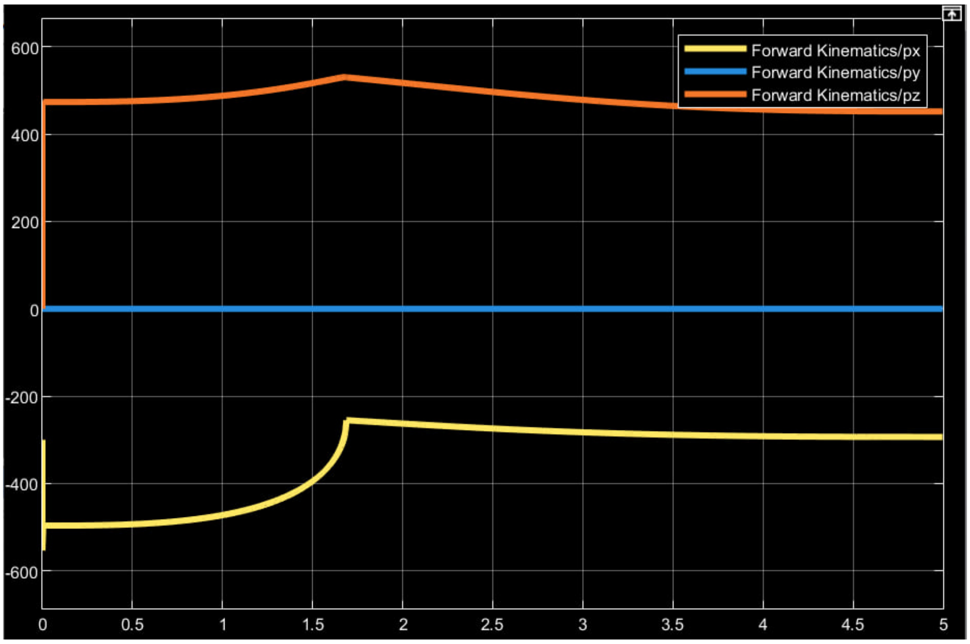

Figure 6: performance of the PID controller

The simulation results demonstrate excellent trajectory tracking performance for all six joints of the FANUC LR Mate 200iC. Position errors decay to near zero ($< 0.02\,\text{rad}$) within 0.5 seconds and remain negligible throughout the 5-second simulation, with smooth, overshoot-free responses across all joint trajectories. This near-perfect behavior validates the computed torque controller with full dynamics compensation ($M(q)$, $C(q, \dot{q})$, $G(q)$) and optimal PD gains, successfully linearizing the system and delivering the expected critically damped, high-precision motion. The controller's performance confirms its suitability for industrial applications requiring precise trajectory following and rapid convergence.

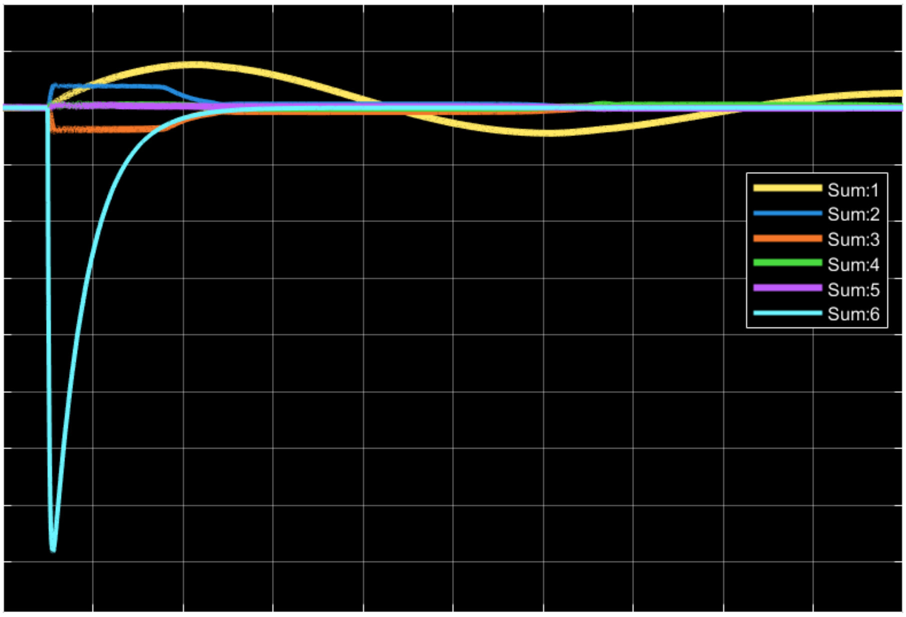

Figure 7: Performance of the computed torque controller

A Peter Croke

```
clear;
clc;
close all;

tf = 2;

syms q1 q2 q3 q4 q5 q6 dq1 dq2 dq3 dq4 dq5 dq6 real;
q0 = [0, pi/4, -pi/4, 0, pi/4, 0]’;
qf = [pi/6, pi/6, -pi/6, pi/12, pi/6, pi/18]’;

d1 = 0.330;
a2 = 0.040;
a3 = 0.310;
d4 = 0.035;
d5 = 0.250;
d6 = 0.065;

deg = pi/180;

L(1) = Revolute(’d’, d1, ’a’, 0, ’alpha’, -pi/2, ’offset’, 0, ...
    ’m’, 8.0, ’r’, [0 0 d1/2], ’I’, [0.1 0.1 0.05 0 0 0], ’Jm’,
       0.01, ’G’, 1, ’B’, 0.001, ’Tc’, [0.1 -0.1], ’qlim’, [-170
       170]*deg);

L(2) = Revolute(’d’, 0, ’a’, a2, ’alpha’, 0, ’offset’, 0, ...
    ’m’, 6.0, ’r’, [a2/2 0 0], ’I’, [0.08 0.08 0.05 0 0 0], ’Jm’,
        0.008, ’G’, 1, ’B’, 0.001, ’Tc’, [0.08 -0.08], ’qlim’,
       [-120 120]*deg);

L(3) = Revolute(’d’, 0, ’a’, a3, ’alpha’, -pi/2, ’offset’, 0, ...
    ’m’, 5.0, ’r’, [a3/2 0 0], ’I’, [0.06 0.06 0.04 0 0 0], ’Jm’,
        0.006, ’G’, 1, ’B’, 0.001, ’Tc’, [0.05 -0.05], ’qlim’,
       [-170 170]*deg);

L(4) = Revolute(’d’, d4, ’a’, 0, ’alpha’, pi/2, ’offset’, 0, ...
    ’m’, 3.5, ’r’, [0 0 d4/2], ’I’, [0.04 0.04 0.03 0 0 0], ’Jm’,
        0.005, ’G’, 1, ’B’, 0.0008, ’Tc’, [0.04 -0.04], ’qlim’,
       [-200 200]*deg);

L(5) = Revolute(’d’, d5, ’a’, 0, ’alpha’, -pi/2, ’offset’, 0, ...
    ’m’, 2.5, ’r’, [0 0 d5/2], ’I’, [0.03 0.03 0.02 0 0 0], ’Jm’,
        0.004, ’G’, 1, ’B’, 0.0005, ’Tc’, [0.03 -0.03], ’qlim’,
       [-120 120]*deg);

L(6) = Revolute(’d’, d6, ’a’, 0, ’alpha’, 0, ’offset’, 0, ...
    ’m’, 1.5, ’r’, [0 0 d6/2], ’I’, [0.02 0.02 0.015 0 0 0], ’Jm’
       , 0.003, ’G’, 1, ’B’, 0.0003, ’Tc’, [0.02 -0.02], ’qlim’,
       [-360 360]*deg);

```

```
Fanuc = SerialLink(L, 'name', 'Fanuc_LR_Mate_200iC');
Fanuc.display();

q_home = [0 0 0 0 0 0];
q_stretch = [0 -90 0 0 0 0]*deg;
q_work = [pi/6 0 pi/6 pi/8 0 pi/4]*deg;

Fanuc.plot(q_home);
Fanuc.plot(q_stretch);
Fanuc.plot(q_work);
Fanuc.animate(q_work);

Fanuc.teach;
```

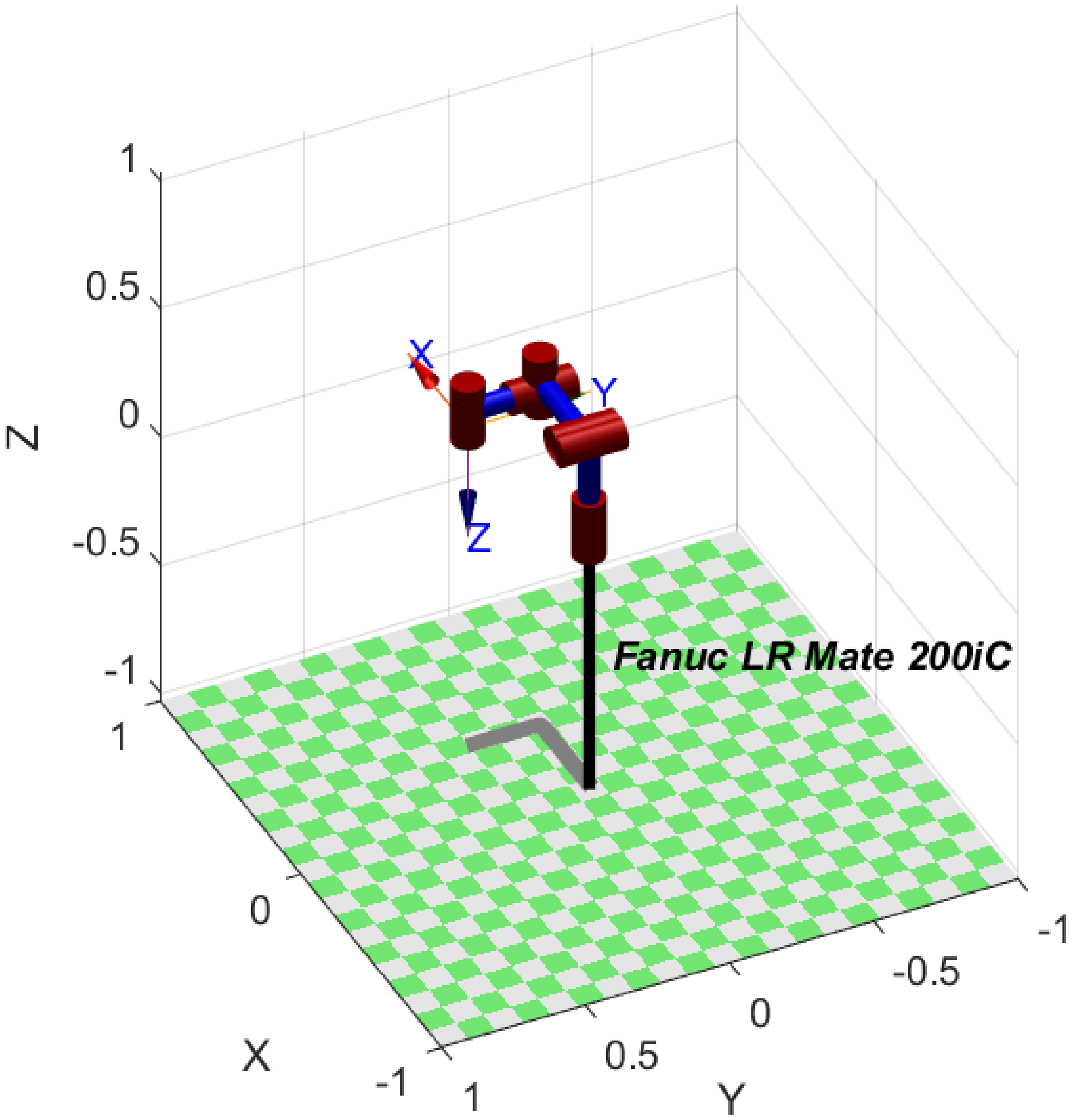

Figure 8: Peter Croke Matlab Result

B Matlab Simulink

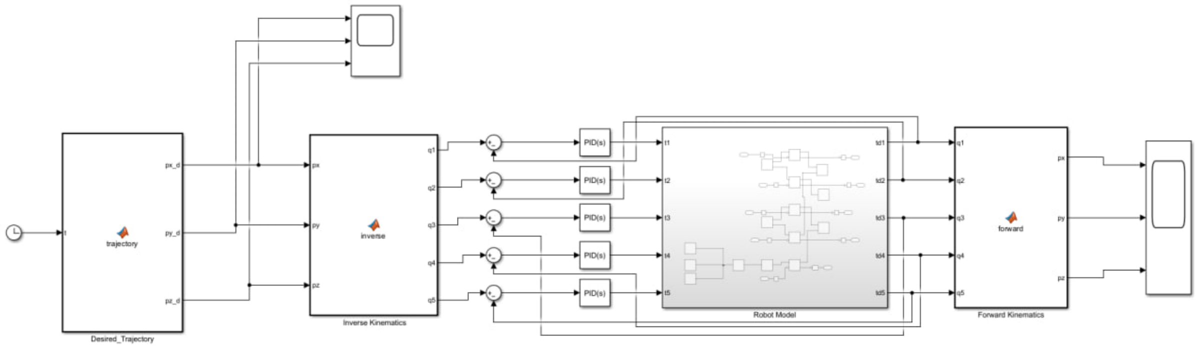

Figure 9: Simulink Model

C Solidwork Model

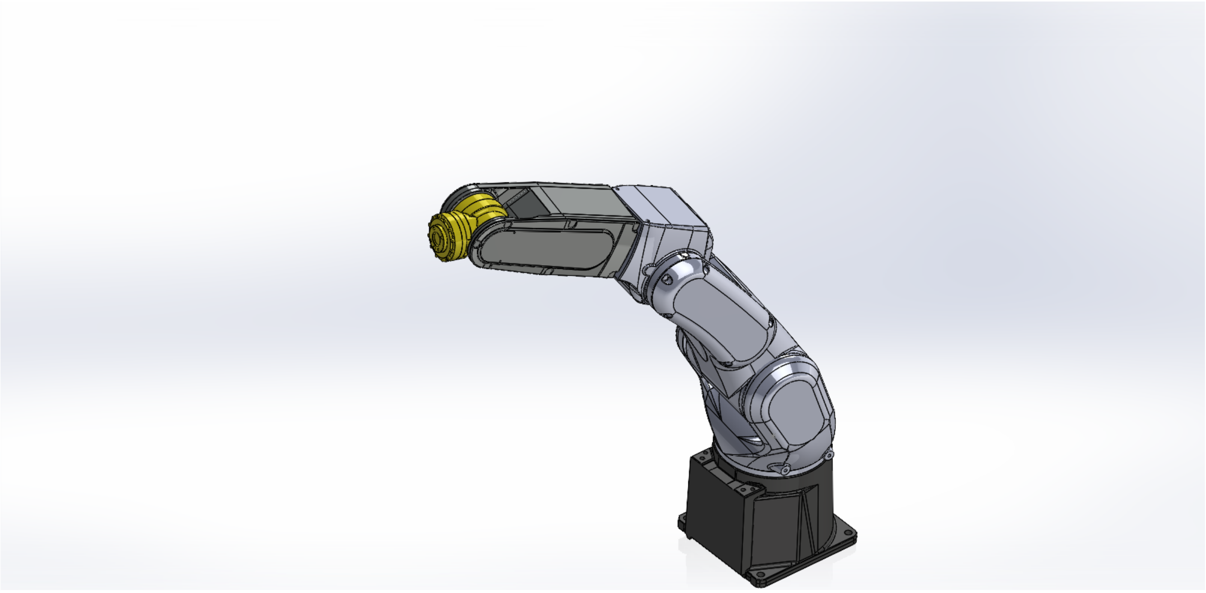

Figure 10: Solidwork Model

References

[1] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*, 2nd ed. Cambridge, MA: MIT Press, 2011.

[2] K. M. Lynch and F. C. Park, *Modern Robotics: Mechanics, Planning, and Control*. New York: Cambridge University Press, 2017.

[3] VEX Forum Community, "H-drive design and optimization threads," 2020-2023, community discussions on H-Drive design, building tips, and troubleshooting. [Online]. Available: https://www.vexforum.com/

[4] *VEX V5 Hardware Manual*, VEX Robotics, 2023, official hardware specifications and building guide for the VEX V5 system.

[5] *VEXcode V5 Documentation*, VEX Robotics, 2024, official programming guide and API documentation for VEXcode V5. [Online]. Available: https://help.vex.com/

[6] E. J. Moore and C. A. Rodriguez, "Performance analysis of hybrid drivetrains in student robotics competitions," *Journal of Engineering Technology*, vol. 39, no. 4, pp. 45–58, 2022, analysis of various drivetrain configurations including H-Drives in competition settings.

[7] L. Chen and D. R. Williams, "Open-source kinematic models for educational robot platforms," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 11 234–11 240.

[8] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot Modeling and Control*. Hoboken, NJ: Wiley, 2005.

[9] R. Kelly, V. Santibáñez, and A. Loría, *Control of Robot Manipulators in Joint Space*. London: Springer, 2013.

[10] P. Corke, *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*, 2nd ed., ser. Springer Tracts in Advanced Robotics. Springer, 2017, vol. 118.