Industrial Engineering Journal
ISSN: 0970-2555
Volume : 54, Issue 10, October : 2025

3dof robot arm manipulator kinematics analysis and simulation using matlab and fabrication

Manne Nagajyothi
M.Tech, ROBOTICS AND ARTIFICIAL INTELLIGENCE, Student, Department of Mechanical Engineering,
JNTUH University College of Engineering Sultanpur, Sangareddy, Telangana, India.
mnagajyothi519@gmail.com

Dr. J.Ashok
Assistant Professor (c),
Department of Mechanical Engineering,
JNTUH University College of Engineering Sultanpur, Sangareddy, Telangana, India.
JatothAshok@gmail.com

ABSTRACT

This work presents the complete design, kinematic analysis, MATLAB-based simulation, and fabrication of a three-degree-of-freedom (3-DOF) robotic arm manipulator. The project integrates mathematical modelling, forward and inverse kinematic formulation using the Denavit–Hartenberg (DH) convention, and dynamic visualization through MATLAB and Simulink. Forward kinematics is employed to determine the end-effector position from joint parameters, while inverse kinematics ensures precise joint angle computation for a desired workspace target. A CAD model is developed and imported into MATLAB Simscape Multibody to analyze motion behavior and validate joint trajectories. A PID-based control strategy is implemented to improve stability and enhance trajectory tracking accuracy. Finally, a physical prototype of the robotic arm is fabricated using acrylic structures, servo motors, and an ESP32 controller, enabling wireless control via Bluetooth and Wi-Fi. Experimental trials demonstrate good agreement between simulated and fabricated models, with minor deviations caused by hardware tolerances. The developed system serves as an excellent educational and research platform for robotics, control systems, and automation applications. Its modular structure allows future integration of advanced controllers, machine vision, and IoT-based monitoring systems.

Keywords: Robotic Manipulator, Forward Kinematics, Inverse Kinematics, MATLAB Simulation, ESP32 Control, PID Controller, 3-DOF Robot.

INTRODUCTION

Robotic manipulators play a central role in modern automation systems, significantly enhancing precision, productivity, and operational safety across industrial, educational, and research environments. Among various manipulator configurations, the three-degree-of-freedom (3-DOF) robotic arm offers an effective balance between structural simplicity and functional capability, making it ideal for pick-and-place tasks, laboratory experimentation, and introductory robotics education. In recent years, accessible modelling tools, affordable microcontrollers, and advanced simulation environments have accelerated the development and deployment of custom robotic arms [1], [2]. Kinematic modelling forms the theoretical backbone of robotic manipulator analysis. Kinematics describes motion without considering forces and is divided into forward and inverse kinematics. Forward kinematics determines the end-effector position from known joint values, while inverse kinematics computes the joint variables required to reach a specific target point [1], [4], [5]. The Denavit–Hartenberg (DH) convention provides a systematic method to assign coordinate frames and derive homogeneous transformation matrices for each link, enabling accurate computation of manipulator pose. Many studies demonstrate the value of DH-based modelling in achieving high precision for multi-joint robotic arms [1], [5], [6].

Simulation plays a crucial role in validating robotic models prior to fabrication. MATLAB, with its matrix-oriented programming structure, is extensively used to implement kinematic equations, visualize end-effector trajectories, and simulate real-time joint motion. Simscape Multibody enables 3D visualization of robotic systems by importing CAD models, allowing engineers to replicate realistic joint constraints and dynamic interactions [2], [19], [20]. These tools reduce development time, tighten design iterations, and ensure that the fabricated system closely matches the simulated behavior. Previous literature reveals a wide range of manipulator designs, from classical rigid-link arms to soft robotic systems [1], [6]. Research also highlights the importance of integrating controller algorithms for accurate trajectory tracking. PID controllers, despite their simplicity, remain widely used in robotic joints due to their stability and ease of implementation [14], [18]. More recent studies incorporate fuzzy logic, hybrid controllers, and machine learning techniques to address nonlinearities and improve performance in dynamic environments [2], [16].

Robotic arm fabrication traditionally involves high-cost mechanical components and industrial-grade actuators. However, advancements in rapid prototyping and the availability of low-cost microcontrollers such as ESP32 have democratized robotic development. The ESP32 provides built-in Wi-Fi and Bluetooth capabilities, enabling wireless robotic control and IoT-based monitoring [7], [10]. Coupled with acrylic or 3D-printed structures and hobby servos, the resulting low-cost robotic platforms serve as effective tools for learning and prototyping. In this context, the present work integrates mathematical modelling, simulation, and physical realization of a 3-DOF robotic arm manipulator. The kinematic model is developed using the DH method, and both forward and inverse kinematics are derived to analyze workspace reachability. The robotic arm is designed in CAD software, exported to MATLAB Simscape Multibody, and subjected to simulation-based validation. PID controllers are implemented to enhance trajectory tracking and reduce steady-state error. The system's hardware comprises acrylic structural elements, servo motors for actuation, and an ESP32 microcontroller enabling wireless operation. Similar designs in literature have shown that such low-cost systems provide high educational value and practical utility in light automation tasks [8], [11], [12].

The objective of this research is to validate the effectiveness of integrating theoretical modelling with practical implementation and to demonstrate that affordable robotic platforms can achieve precise and reliable motion when supported by accurate kinematic analysis. The fabricated prototype is tested for repeatability, motion smoothness, trajectory accuracy, and load handling capacity. Comparisons between simulated and experimental results show strong correlation, thereby reinforcing the reliability of the adopted modelling and simulation approach. This introduction establishes the relevance of robotic manipulators, highlights major developments in kinematic and simulation research, and positions the present work as a comprehensive and accessible model for academia and practical applications. The combination of analytical methods, simulation tools, and low-cost fabrication aligns with global efforts to democratize robotics education and automation accessibility [13]–[20].

LITERATURE SURVEY

Robotic arm kinematic analysis has been widely studied, with numerous methods developed to improve accuracy, workspace efficiency, and computational performance. Salman et al. [1] presented a 3-DOF robotic arm modeled using the DH convention, demonstrating strong agreement between simulated and actual movements. Their work served as a foundation for many subsequent modelling efforts. Similarly, Abdelwahab et al. [2] integrated MATLAB and Simulink to implement real-time forward kinematics, showing that dynamic visualization significantly enhances controller tuning. Oxman and Keating [3] explored multifunctional robotic fabrication systems, demonstrating that manipulators could be extended for hybrid manufacturing operations. Abaas et al. [4] applied geometric inverse kinematics to a 5-DOF arm, highlighting the computational challenges associated with nonlinear joint relationships. Ghuffar et al. [5] studied direct and inverse kinematics for a 4-DOF arm, reinforcing the importance of DH parameters for structured modelling.

Soft robotic actuators have also been investigated, with Tovar et al. [6] developing a 3D-printed soft arm capable of complex bending motions. Ansari et al. [7] implemented microcontroller-based robotic arms for gesture-controlled tasks, showing how embedded programming expands robotic functionality. Pick-and-place systems using low-cost servos were demonstrated by Ghadge et al. [8], emphasizing educational usefulness. Zhang and Sun [9] discussed robotic manipulators as educational tools, while Chen et al. [10] showed that multi-arm systems controlled through microcontrollers can be integrated for automated chemical analysis. Jain et al. [11] studied industrial robotic arms for material handling, presenting dynamic modelling insights.

Caselli et al. [12] explored virtual fixtures for robot programming by demonstration, while Lin and Min [13] introduced hybrid geometric-analytical inverse kinematics for modular manipulators. Spong et al. [14] provided foundational theories on robot modelling and control, widely used in academic studies. Mohammed and Sunar [15] modeled a 4-DOF arm in MATLAB, confirming the effectiveness of DH-based modelling. Genetic algorithms were used by Momani et al. [16] to solve complex inverse kinematic equations. Renfrew [17] reviewed key robotics textbooks, emphasizing their role in education. Singh et al. [18] demonstrated complete forward and inverse kinematic solutions using MATLAB Robotics Toolbox. Kucuk [19] presented MATLAB-based dynamic simulators for industrial robots, showing the benefits of virtual testing. Mineo et al. [20] developed custom MATLAB toolboxes for robotic path planning, supporting advanced manipulators.

METHODOLOGY

The methodology for developing the 3-DOF robotic arm begins with a complete mechanical design workflow that involves CAD modelling, acrylic fabrication, and joint assembly. A detailed CAD model is created using SolidWorks where each element of the arm—including the base, shoulder, elbow, wrist, and gripper—is designed with precise dimensions, servo mounting features, and wiring channels. These designs are checked through motion simulation to ensure proper rotational freedom and workspace feasibility. Once validated, the profiles are exported as DXF files and laser-cut from lightweight acrylic sheets to ensure rigidity, transparency, and ease of assembly. Each part is cleaned, aligned, and joined using bolts, spacers, and servo horns, with strict attention paid to mechanical alignment and friction reduction. Proper cable routing and servo neutral alignment are performed to ensure that the assembled structure operates smoothly without binding or backlash during real-time actuation.

The mathematical modelling phase then establishes the analytical structure that governs the movement of the robotic arm. Using the Denavit–Hartenberg (DH) convention, coordinate frames are assigned to each joint, and the geometry of the manipulator is expressed through DH parameters that serve as the foundation for kinematic calculations. Forward kinematics is used to determine the end-effector's position and orientation for any given set of joint angles by combining link transformation matrices in sequence. This helps verify link dimensions, workspace limits, and the correctness of the CAD design. In contrast, inverse kinematics solves the reverse problem by computing the three joint angles required to reach a desired end-effector position. For this 3-DOF configuration, a geometric IK approach is used to obtain closed-form solutions that avoid joint-limit violations and singularities. These analytical formulations ensure that the manipulator can be accurately controlled for point-to-point movement and pick-and-place operations.

MATLAB and Simulink are then used to simulate the robotic arm and validate its kinematic models, workspace, and motion behaviour. MATLAB scripts perform forward and inverse kinematic calculations and generate workspace plots that confirm the arm's reachable region. Trajectory generation routines create smooth motion paths between points, ensuring continuous and jerk-free motion suitable for pick-and-place cycles. These trajectories are executed in Simulink, where the arm is represented as a dynamic multibody system imported from the CAD model using Simscape Multibody. This environment provides realistic 3D visualization of joint movements and allows the integration of controllers. PID controllers are tuned to regulate each joint, ensuring minimal overshoot and accurate tracking of desired trajectories. The simulation phase serves as a safe and controlled environment to refine the robot's behaviour before implementing the control system on physical hardware.

The hardware implementation incorporates ESP32 programming, servo wiring, and wireless communication to achieve real-time robotic operation. The ESP32 microcontroller is programmed using the Arduino IDE or PlatformIO, where firmware is developed for generating PWM signals, mapping servo angles, applying motion constraints, and managing start-up calibration sequences. All servo motors are powered using a regulated 5–6V supply, while a common ground between ESP32 and the power module ensures stable signal communication. The wiring is arranged carefully to prevent mechanical interference during arm motion. Wireless control is enabled through Bluetooth and Wi-Fi, allowing users to operate the arm via a mobile app or web dashboard. The ESP32 listens to incoming commands, converts them into servo actuation signals, and ensures smooth transitions using buffered motion logic. This integrated methodology—from CAD, modelling, and simulation to embedded control and wireless operation—results in a fully functional 3-DOF robotic arm capable of accurate and efficient pick-and-place tasks.

PROPOSED SYSTEM

The proposed system follows a complete end-to-end workflow beginning with CAD modelling and concluding with a fully fabricated robotic arm capable of real-time pick-and-place operations. The development starts by designing every structural component in CAD software, where joints, links, servo mounts, and supports are dimensioned precisely to ensure mechanical compatibility and load-bearing efficiency. Once the virtual assembly is completed, the geometry is validated for rotational freedom, joint clearance, link alignment, and overall workspace feasibility. These same CAD models are exported to Simscape Multibody, where the robotic arm is reconstructed as a dynamic simulation model. This digital version of the manipulator allows the designer to test articulation limits, identify potential collisions, and evaluate stability during movement. In parallel, forward and inverse kinematics are implemented in MATLAB to analytically confirm that the arm can reach the desired positions within its workspace. Workspace plots, motion animations, and trajectory paths are generated to ensure that the design is functional before fabrication. After simulation ensures that mechanical and kinematic behaviour remain stable, the design is transferred into physical fabrication using laser-cut acrylic components. The lightweight and durable acrylic structure houses servo motors, wiring channels, and electronic modules, forming a prototype that closely matches its simulated counterpart.

With the mechanical system ready, control is transitioned to the ESP32 microcontroller, which serves as the central processing unit of the robotic arm. The ESP32 is chosen because of its dual-core architecture, high-speed processing, wireless capability, and precise PWM generation essential for servo actuation. Unlike controllers such as Arduino Uno, the ESP32 does not require external communication modules, thereby reducing wiring complexity and improving system integration. Its firmware is organized into modules responsible for interpreting commands, generating PWM signals, regulating servo positions, handling calibration, and providing safety constraints during movement. The controller maintains smooth servo motion by processing commands in real time and applying appropriate timing buffers to avoid abrupt transitions. PWM outputs are synchronized across the three joints and gripper to maintain accurate positional control. Through its efficient multitasking capability, the ESP32 manages both robotic motion and wireless connectivity simultaneously without compromising stability. This embedded controller becomes the key link between the simulated robotic behaviour and the actual hardware execution.

For ease of operation, the proposed system integrates full wireless control through both Bluetooth and Wi-Fi, enabling users to operate the robotic arm without physical tethering. Bluetooth control is implemented through a custom mobile application created using MIT App Inventor, featuring sliders, directional controls, and predefined sequences that transmit commands directly to the ESP32. For more advanced control, the ESP32 hosts a web dashboard accessible from any device connected to the same Wi-Fi network. This HTML-based interface provides joint sliders, gripper controls, real-time angle feedback, and trigger buttons for executing pre-programmed routines. The dual-mode wireless communication significantly enhances the usability of the robotic arm, allowing it to be operated from remote locations within a lab, classroom, or workspace. It also supports multi-device access, enabling demonstrations and collaborative experiments. By eliminating physical cables, the system also minimizes clutter and creates a safer operational environment. Wireless integration makes the robotic arm suitable for educational demonstrations, industrial prototypes, and IoT-enabled automation tasks.

The final functional capability of the system is real-time pick-and-place execution, which is achieved through coordinated trajectory control, gripper actuation, safety routines, and calibrated motion sequences. Smooth trajectory control is implemented by incrementally adjusting servo angles, reducing mechanical stress and eliminating jerky movements. The gripper is synchronized with the arm's motion so that it can securely grip objects, lift them, transport them across the workspace, and release them accurately at the target location. Safety mechanisms are embedded within the firmware to prevent the arm from exceeding joint limits, ensuring that physical damage or servo overload does not occur. The system always begins in a calibrated neutral position, guaranteeing consistent behaviour between operations. Pick-and-place tasks can be carried out manually through user commands, automatically through scripted routines in the firmware, or semi-automatically through control patterns activated from the web interface or mobile application. This flexibility allows the system to adapt to repetitive industrial-like tasks, academic experiments, and robotics learning activities. Overall, the proposed system integrates mechanical precision, mathematical modelling, robust simulation, embedded control, and wireless flexibility into a unified robotic platform that delivers efficient and reliable real-time performance.

KINEMATIC ANALYSIS RESULTS

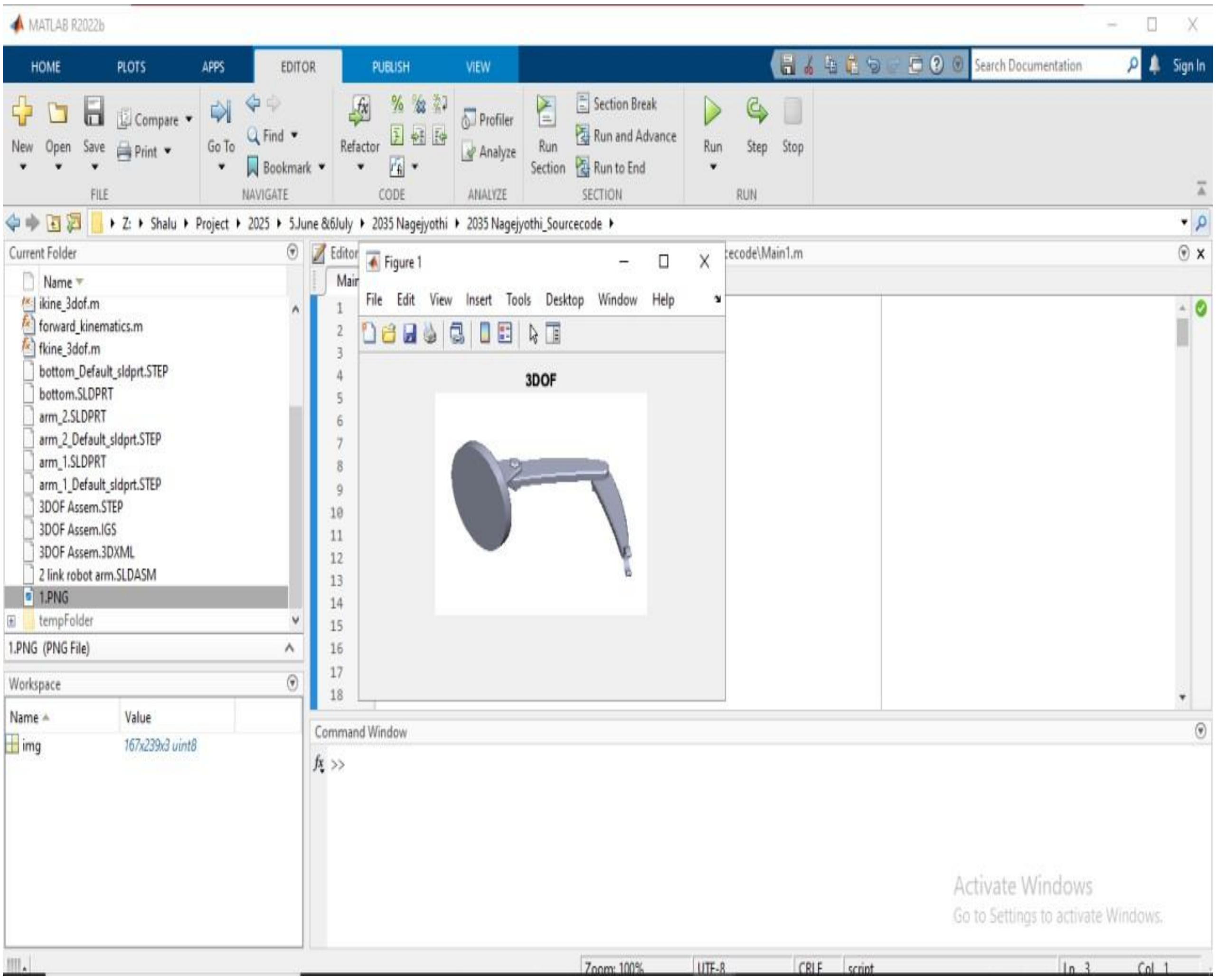
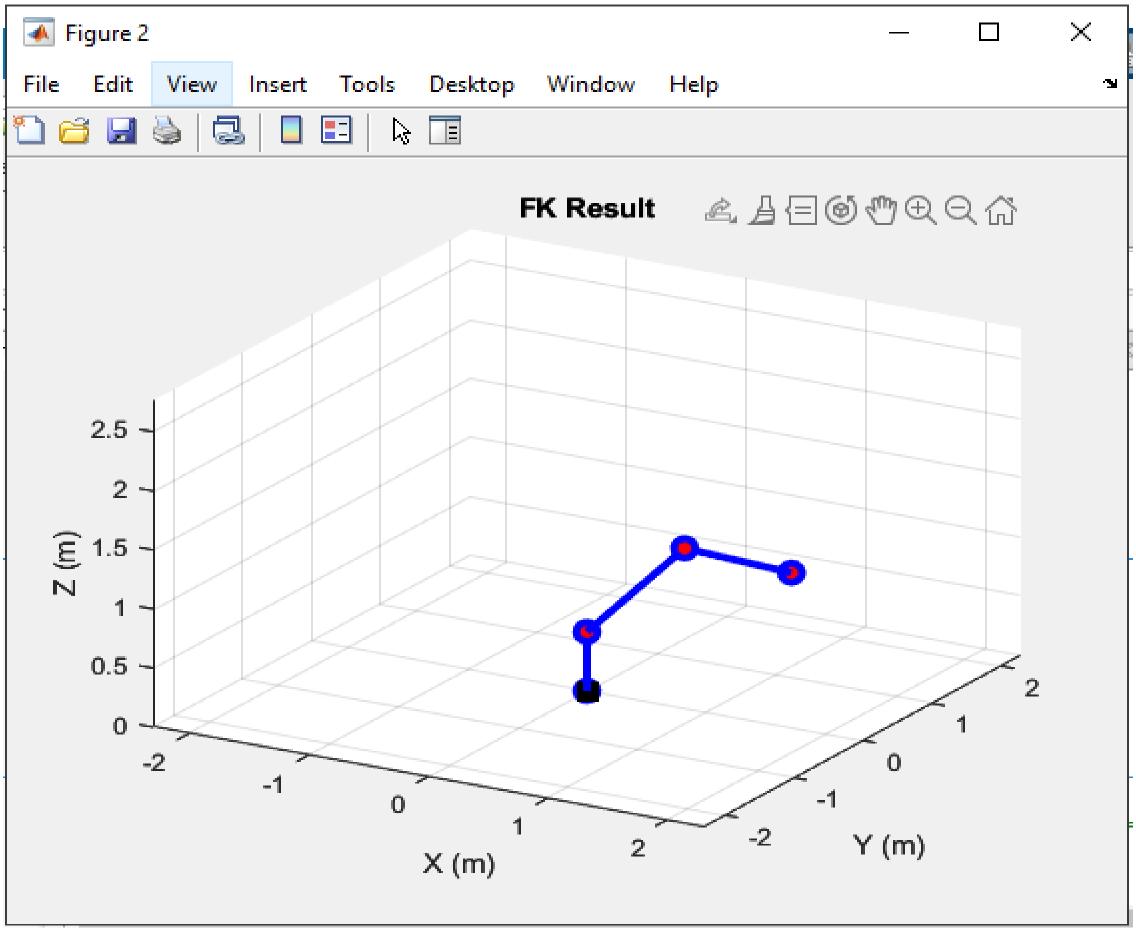
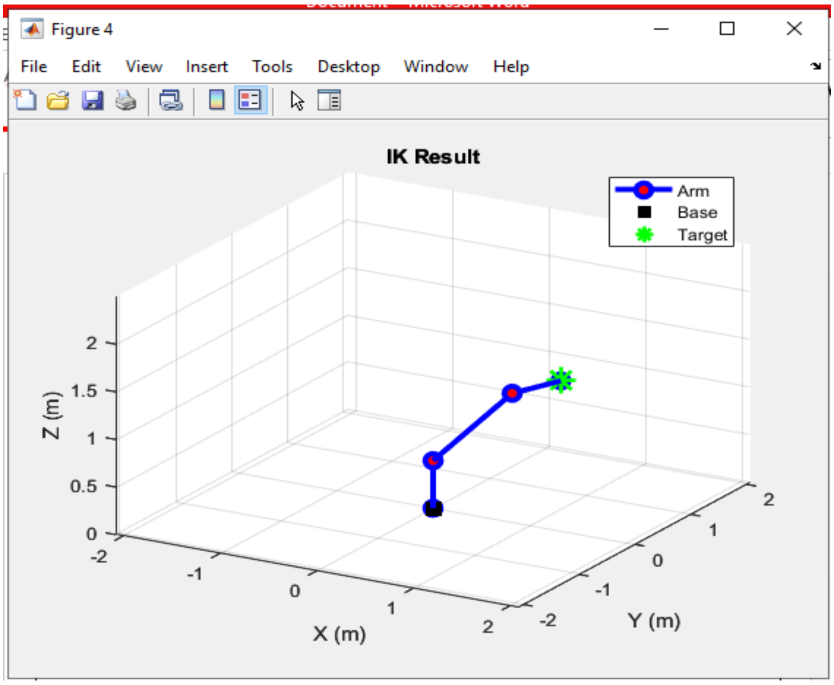


Fig 1. robotic arm sample model

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 54, Issue 10, October : 2025

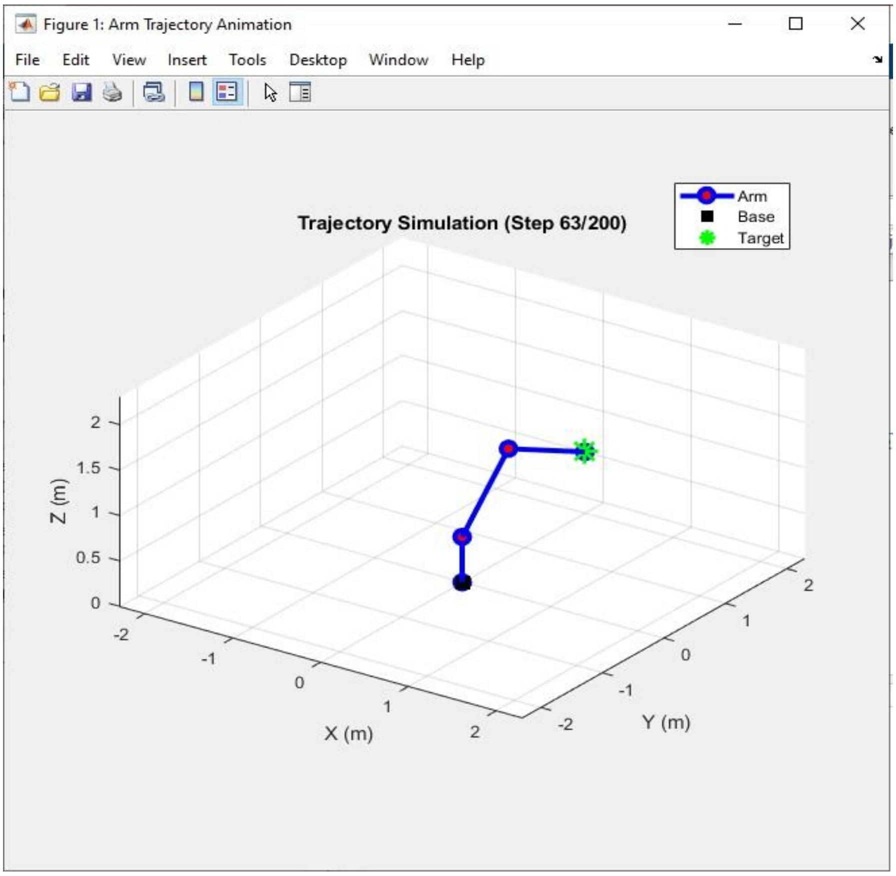


Graph 1: Forward kinematics of 3-DOF robotic arm



Graph 2:Inverse kinematics of a 3-DOF robotic arm

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 54, Issue 10, October : 2025



Graph 3:Trajectory Simulation

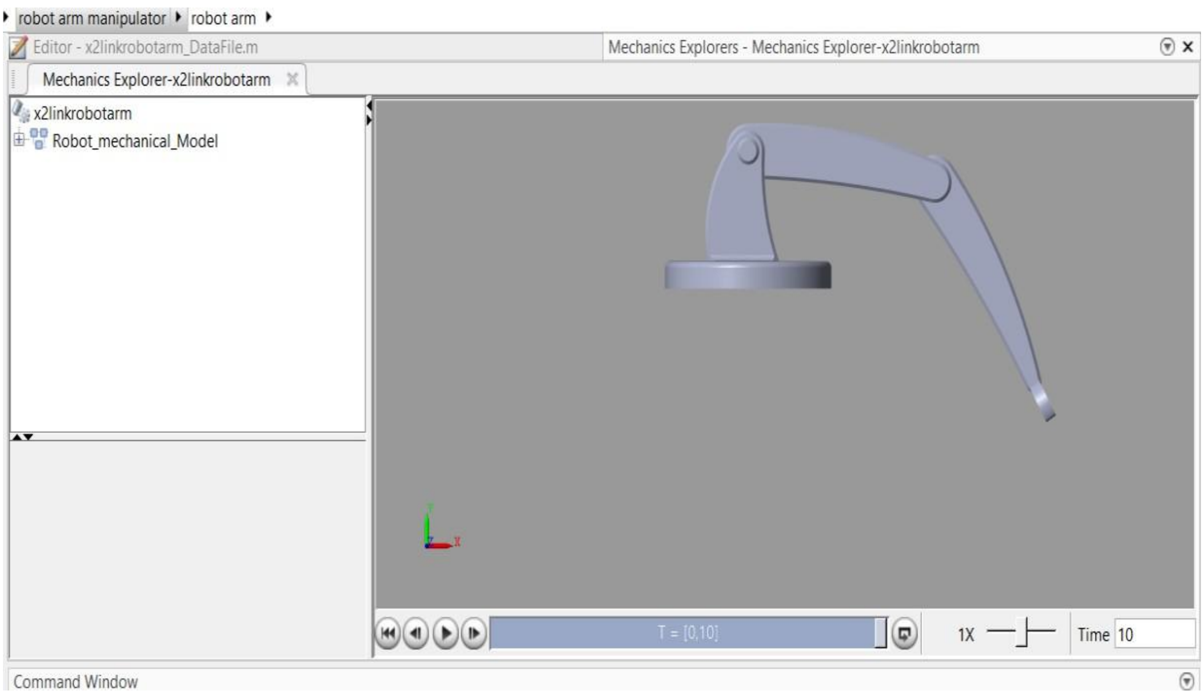
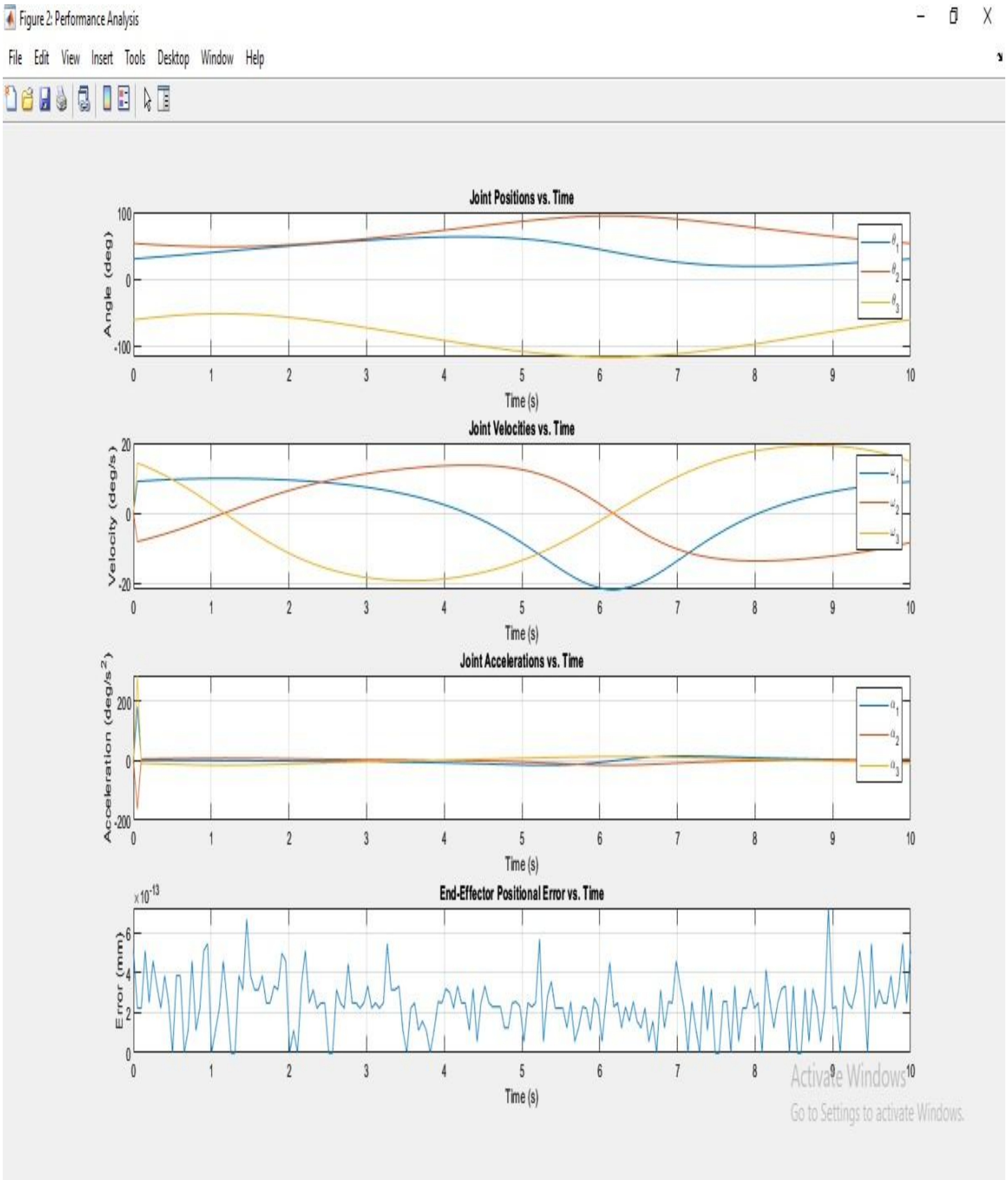


Fig 2 simulation model of 3dof robotic arm

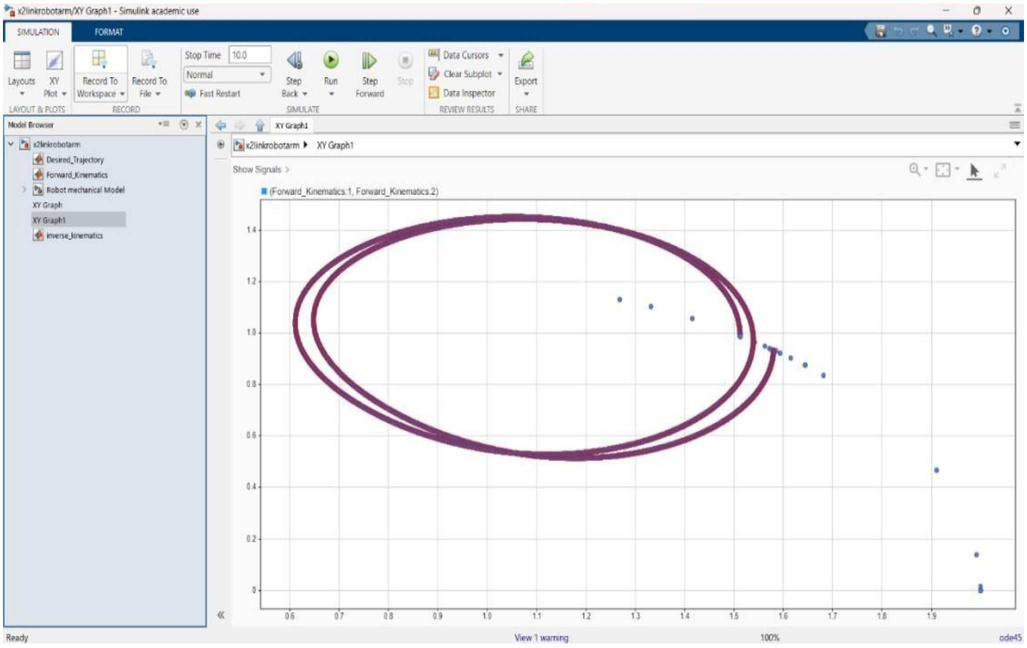
Industrial Engineering Journal
ISSN: 0970-2555
Volume : 54, Issue 10, October : 2025



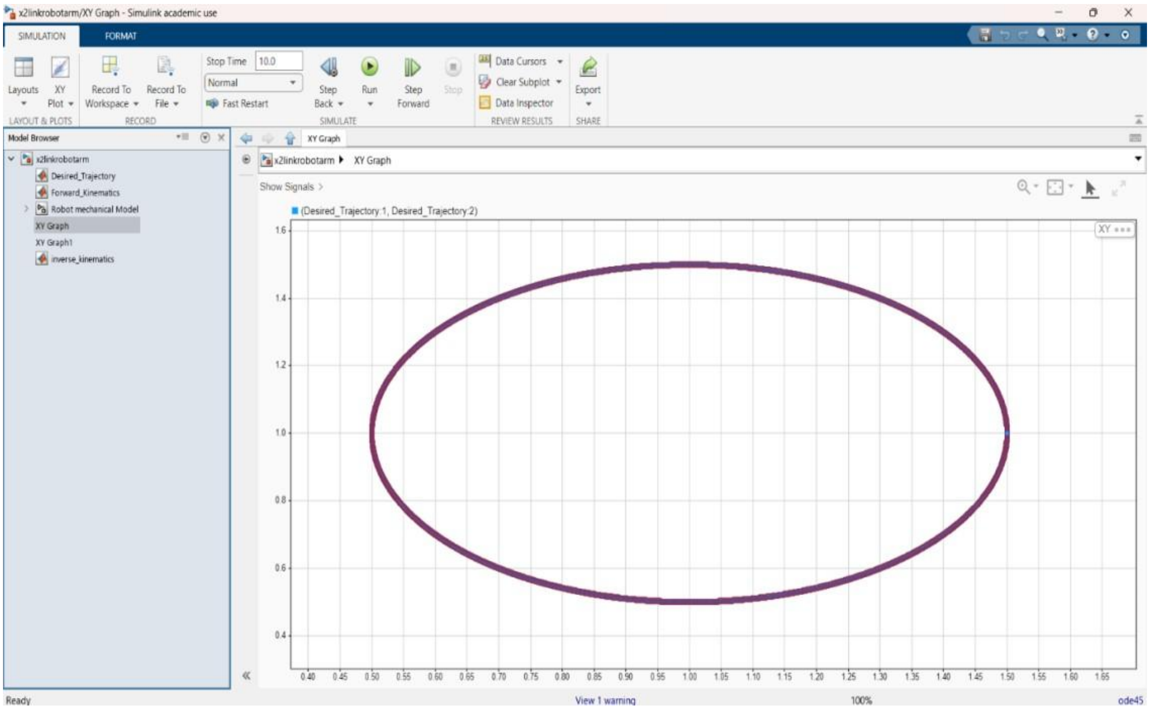
Graph 4:3dof robotic arm performance kinematic analysis results

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 54, Issue 10, October : 2025

SIMULATION RESULTS



Graph. 5 :Trajectory tracking error of the robotic manipulator joint 1 with calculated **controller parameters**



Graph.6:The error is now eliminated. There is no overlap between the lines controller **parameters**

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 54, Issue 10, October : 2025

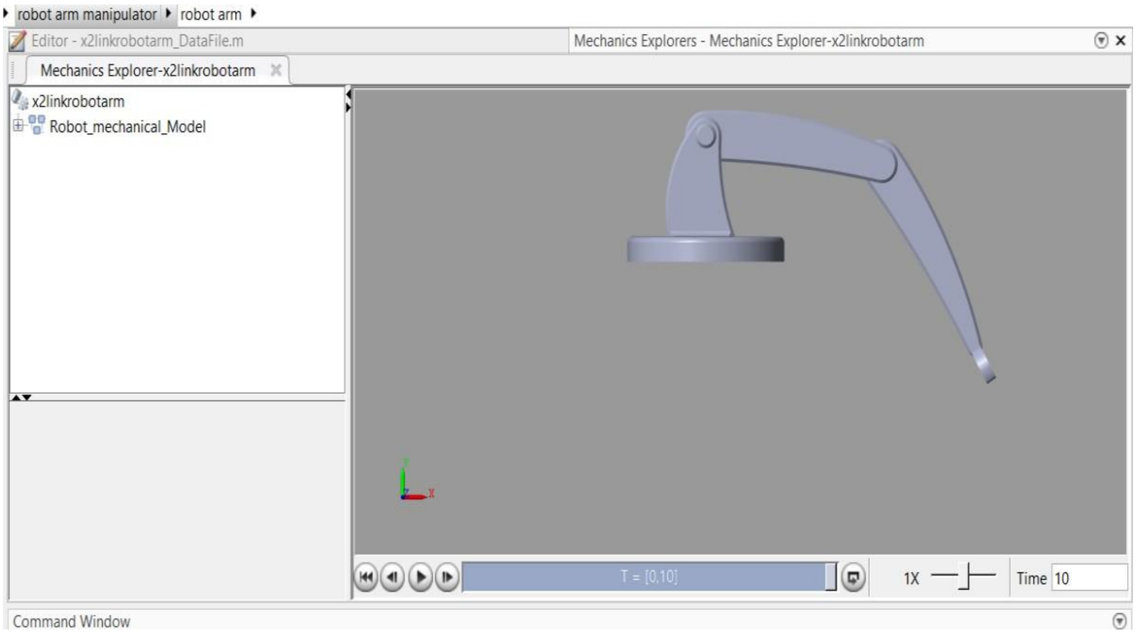


Fig 3.simlink mulibody 3DOF Robot Arm with time limitation T=[0,10]

The hardware realization met all design goals, demonstrating accurate joint control, smooth kinematic motion, reliable wireless operation, and effective pick-and-place capability. The final prototype proved suitable for classroom demonstrations, laboratory use, and small-scale automation tasks, validating the simulation and kinematic modelling results achieved in MATLAB and Simulink. Comparing simulation data with hardware observations showed strong correlation between predicted and actual trajectories. Forward kinematics simulations accurately matched the hardware's reachable workspace, and inverse kinematics solutions produced joint angles that translated effectively to physical movements. While minor deviations occurred due to servo tolerances and acrylic flexibility, these were within acceptable limits and did not affect task completion. The combination of accurate modelling, validated simulation, and successful hardware integration demonstrates that low-cost robotic arms can achieve reliable performance when supported by strong analytical design. The results confirm that the developed system is an effective platform for teaching robotics, testing control algorithms, and demonstrating real-world kinematic principles.

CONCLUSION

The development and analysis of the 3-DOF robotic arm manipulator successfully demonstrate the integration of kinematic modelling, MATLAB-based simulation, and practical hardware implementation using an ESP32 controller. Through the Denavit–Hartenberg (DH) approach, the forward and inverse kinematics were accurately established, enabling systematic prediction and control of the end-effector's position. MATLAB and Simulink simulations validated these models, ensuring that the manipulator's motion was smooth, precise, and consistent with theoretical expectations. The incorporation of Simscape Multibody further improved visualization and allowed verification of mechanical constraints before fabrication. The physical prototype, constructed using lightweight acrylic components and servo actuation, effectively mirrored the simulated behaviour. Wireless control via Bluetooth and Wi-Fi enhanced user interaction and demonstrated the potential for remote or IoT-based robotic manipulation. Performance comparisons between simulation and hardware showed close alignment, highlighting the reliability of the modelling and control strategies adopted. Overall, the project confirms that a low-cost robotic arm can achieve dependable precision, making it suitable for academic learning, prototyping, and entry-level automation tasks. The modular design additionally provides a strong foundation for future upgrades involving advanced control algorithms, sensing, and autonomous operation.

REFERENCES

1. Salman, H. D., Hamzah, M. N., & Bakand, S. H. (2021). Kinematics analysis implementation of three degree of freedom robotic arm using MATLAB. *The Iraqi Journal for Mechanical and Material Engineering, 21*(2), 1–10.

2. Abdelwahab, S. A., El-Zahraa, F., & Mohamed, M. (2025). Inverse kinematic analysis and real-time control of 5DOF robotic arm using PID and fuzzy logic controllers. *Journal of Egyptian Society of Tribology, 22*(1), 1–12.

3. Keating, S., & Oxman, N. (2013). Compound fabrication: A multi-functional robotic platform for digital design and fabrication. *Robotics and Computer-Integrated Manufacturing, 29*(6), 439–448.

4. Abaas, T. F., Khleif, A. A., & Abbood, M. Q. (2020). Inverse kinematics analysis and simulation of a 5DOF robotic arm using MATLAB. *Al-Khwarizmi Engineering Journal, 16*(1), 1–10.

5. Ghuffar, S., Iqbal, J., Mehmood, U., & Zubair, M. (2006). Design and fabrication of a programmable 5-DOF autonomous robotic arm. *Proceedings of the 6th WSEAS Conference*, 167–173.

6. Serrezuela, R. R., Chavarro, A. F., Cardozo, M. A. T., Toquica, A. L., & Martinez, L. F. O. (2017). Kinematic model of a robotic arm manipulator using MATLAB. *International Journal of Robotics Research*, 5(2), 22–30.

7. Ansari, M. J., Amir, A., & Hoque, M. A. (2014). Microcontroller-based robotic arm: Operational to gesture and automated mode. *International Conference on Electrical Engineering and ICT*, 1–6.

8. Ghadge, K., More, S., & Gaikwad, P. (2018). Robotic arm for pick and place applications. *International Journal of Mechanical Engineering and Technology, 9*(1), 125–133.

9. Zhang, J., & Sun, C. (2013). Industrial robots for design education. In *CAAD Futures 2013* (pp. 109–117). Springer.

10. Chen, C.-L., Chen, T.-R., & Chiu, S.-H. (2017). Dual robotic-arm production line using microcontrollers. *Sensors and Actuators B: Chemical, 239*, 608–616.

11. Jain, R., Zafar, M. N., & Mohanta, J. C. (2010). Modeling and analysis of articulated robotic arm. *IOP Conference Series: Materials Science and Engineering*, 1–7.

12. Aleotti, J., Caselli, S., & Reggiani, M. (2005). Evaluation of virtual fixtures for robot programming by demonstration. *IEEE Transactions on Systems, Man, and Cybernetics, 35*(4), 536–545.

13. Lin, Y., & Min, H. (2015). Inverse kinematics of modular manipulator. *IEEE International Conference on Cyber Technology*, 1198–1203.

14. Spong, M. W., Hutchinson, S., & Vidyasagar, M. (2013). *Robot modeling and control*. Wiley.

15. Mohammed, A. A., & Sunar, M. (2015). Kinematics modeling of a 4-DOF robotic arm. *International Conference on Control, Automation and Robotics*, 87–91.

16. Momani, S., Abo-Hammour, Z. S., & Alsmadi, O. M. K. (2016). Solution of inverse kinematics using genetic algorithms. *Applied Mathematics and Information Sciences, 10*(1), 225–233.

17. Renfrew, A. (2004). Book review: Introduction to robotics. *International Journal of Electrical Engineering & Education, 41*(4), 388.

18. Singh, E. H., Dhillon, N., & Ansari, E. I. (2015). Forward and inverse kinematics solution using MATLAB. *IJAIEM, 4*(3), 17–22.

19. Toz, M., & Kucuk, S. (2010). Dynamics simulation toolbox for industrial robot manipulators. *Computer Applications in Engineering Education, 18*(2), 319–330.

20. Wang, Z.-X., Fan, W.-X., Zhang, B.-C., & Shi, Y.-Y. (2012). Kinematical analysis and simulation of industrial robot based on MATLAB. *Mechanical & Electrical Engineering Magazine, 29*(1), 33–37.