International Journal of Scientific Research & Engineering Trends
Volume 11, Issue 1, Jan-Feb-2025, ISSN (Online): 2395-566X

Development of Robotic Arm Using Arduino

Lingam Lakshmi Vagdevi, Edupuganti Harshitha
Department of Electronics & Communication Engineering
Institute of Aeronautical Engineering, Hyderabad, Telangana, India

Abstract- Innovation in robotic arm control has been sparked by the development of Arduino-based technology, which provides both experts and enthusiasts with an affordable and user-friendly platform. The creation of robot arm control with an Arduino controller is presented in this work. The project entails integrating sensors and Arduino microcontrollers to provide dynamic and accurate control over a robotic arm. Four servo motors—which rotate left, right, front, and back—control the suggested robot. The study lays the groundwork for future developments in this emerging topic by discussing the difficulties faced during the development process and offering solutions. The demonstrated robotic arm control system has the potential to increase access to robotics education and promote automation innovation due to Arduino's broad availability and low cost.

Index Terms- Arduino, Control system, Robotic Arm, Sensors , Servo motor

I. INTRODUCTION

The introduction highlights the growing burden on humans and the necessity for effective solutions while discussing the ubiquitous influence of technology and automation in today's environment. It presents robotics as a major factor in lowering human labor, increasing accuracy, and conserving resources, time, and money. Computer science, mechanical engineering, and electrical engineering are all involved in robotics. Arduino is a programming language that is used to create robotic equipment. The emphasis is on robotic arms, which imitate the motions of human arms and are used in various industries for picking and positioning goods. With its focus on an Internet of Things (IoT)-based robotic arm, the project showcases the importance of IoT in contemporary technology. The project intends to discuss the technological features, difficulties, and uses of robotic arms, highlighting their possible use.

Automated robotic arms rely on sensors and programs to function with little to no operator input. Automation technology has advanced significantly as a result of the combination of mechanical and electronics engineering.

In order to acquire a competitive edge, modern industry is concentrating on computer-based interaction and robotics to increase production and guarantee constant product quality. Because of their increased speed, intelligence, precision, and capacities, modern robots are becoming more and more appropriate for both low mix-high volume and high mix-low volume production. Collaborative robots are capable of working side by side with people, sharing a workspace, and using sensor technologies to function safely in dangerous situations [10]–[15].

Robots are frequently used to perform repetitive, complex, and hazardous activities, which lessens the need for humans. Because of their flawless performance, they have become widely accepted throughout the world. Robots are useful for home automation systems that serve the elderly and physically disabled, in addition to their industrial uses. By allowing sophisticated machine operations to be performed by hand gestures without physical contact, gesture recognition acts as a dynamic interface between users and robots. Gesture-controlled robots increase autonomy in daily tasks and help disabled people integrate into regular job life

II. BLOCK DIAGRAM

Assembling the mechanical construction, attaching servo motors for joint movements, and integrating electronic components are all necessary to create a robotic arm that is controlled by an Arduino board, as seen in Figure 1. Servo motors are attached to the Arduino board, usually using jumper wires, once the robotic arm has been assembled. For the Arduino and the servo motors to be powered, an appropriate power source is necessary. The servo motors are controlled by a bespoke program that specifies the necessary angles for each joint using the Arduino IDE loaded on a computer. Next, a USB cable is used to upload this program to the Arduino board. The robotic arm is then tested and calibrated to make sure it travels precisely at the designated angles. The four servo motors are connected to the Arduino UNO, which consists of MB102 5v/3.3v, power supply and IR sensor.

The use of servo motors, essential parts that enable precise joint movements, enables Arduino-based control of a robotic arm. These servo motors, which are attached to the Arduino board, function as the robotic arm's muscle and react to programmed guidelines that specify each joint's angles and locations. With the help of its intuitive IDE, the Arduino platform enables the development of customized code that converts intended movements into commands that the servo motors can follow. The range of motion of the robotic arm can be adjusted to guarantee precise and well-coordinated movements through meticulous calibration and testing. The servo motors are essential to carrying out these motions because they supply the mechanical force required to articulate the arm's numerous segments. Using an Arduino to control a robotic arm requires a complex interaction between hardware and software, with the power supply being essential to dependable and seamless operation. The Arduino board acts as the robotic arm's brain, carrying out preset commands or reacting to sensors' real-time inputs. A reliable and suitable power source is necessary to power the system. Usually, this entails supplying electricity to the motor drivers, Arduino board, and—above all—the motors that move the robotic arm. The particular components used in the robotic arm determine the voltage and current requirements, thus choosing a power source that satisfies these needs is essential. To control the power distribution, motor drivers or voltage regulators may also be used.

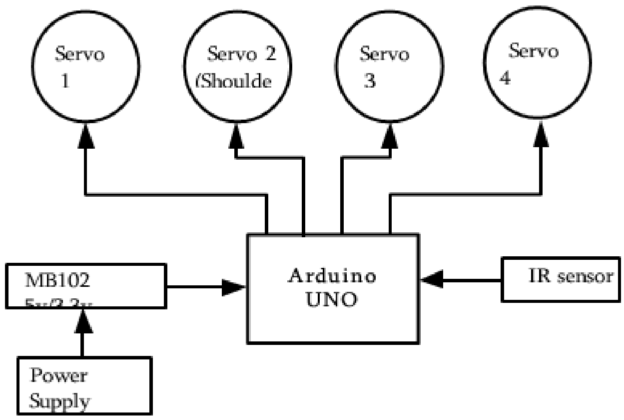

Figure 1. Block diagram interaction of components

In the field of Arduino-controlled robotic arm systems, jumper wires are essential since they act as the glue holding various parts together. These cables, which usually have male and female connectors, let power and signals to flow smoothly between the Arduino board, sensors, motor drivers, and other electronic components in the robotic arm configuration. Jumper wires are essential for creating the connections required between the microcontroller and the motors that move the arm in the context of robotic arm control. They are also used to connect sensors to the Arduino for feedback and control, such limit switches or encoders. Jumper wire color coding makes it easier to differentiate between various functions, expedites the wiring procedure, and improves the system as a whole.

The programming center for controlling and coordinating robotic arms that use Arduino technology is the Arduino integrated development environment (IDE). This intuitive software offers a way to write, compile, and upload code to the Arduino microcontroller, which controls the motion of the robotic arm. Using the Arduino IDE, programmers and enthusiasts can write unique scripts and algorithms that specify the arm's actions, motion patterns, and reactions to outside inputs. Even people with little coding knowledge can use the IDE because it supports the simpler C/C++ programming language. Sensors, motors, and other electronic components can be easily integrated into the control scheme by utilizing the many robotics-specific libraries and functions available inside the Arduino IDE. Furthermore, the IDE offers a serial

To carry out these commands, the microcontroller communicates with a number of different parts. The Arduino receives input from the sensors—such as limit switches or encoders—about the location and condition of the robotic arm. To guarantee precise and regulated movements, this information is essential.

Motor drivers control the power delivered to the motors by receiving signals from the Arduino. By converting digital signals into the appropriate power levels, they provide as a bridge between the microcontroller and the motors. The motors are in charge of powering the robotic arm's many joints and actuators. The motor drivers provide them with power and control signals, which enable accurate movement in response to Arduino commands. The power supply gives the Arduino, sensors, and motor drivers the voltage and current they require.

III. ARDUINO IN TINKER CAD DESIGN

The Arduino is displayed in Tinkercad in Figure 2. Communication with servo motors is essential to accomplishing accurate and well-coordinated movements in Arduino robotic arm control systems. Because servo motors can precisely control angular motion, they are frequently seen in robotic arms.

A pulse width modulation (PWM) signal is used by the Arduino to communicate with servo motors. PWM is a technique for encoding data as pulses, where each pulse's width is associated with a distinct control signal. The Arduino uses its output pins to create PWM signals that are used to drive servo motors. The servo motor's signal input wire receives these PWM signals. Information regarding the intended position or angle of the servo motor shaft is contained in the signal.

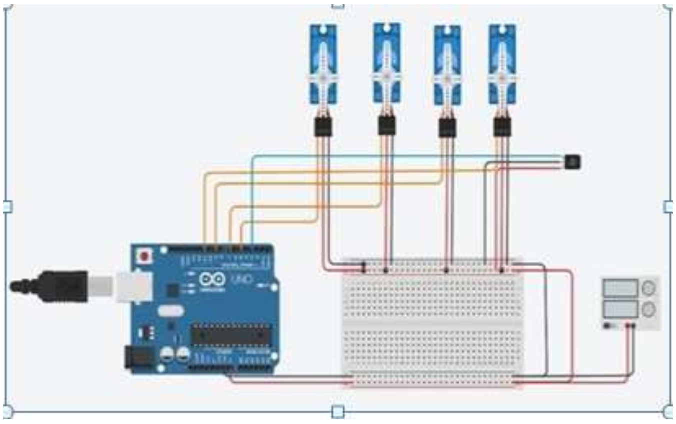

Figure 2. Arduino in Tinkercad

IV. MECHANICAL DESIGN

The arm base waist of the testing robot is seen in Figure 3. The physical framework that establishes the shape and capabilities of a robotic manipulator is known as the robotic arm structure. This structure, which is made up of several interconnected parts, is intended to resemble the suppleness and agility of a human arm.

A robotic arm usually has several joints or links that can move in either a rotational or translational direction. It also frequently has end effectors for certain functions. Actuators—typically motor or servos—connect the joints and propel the arm's motion. The arm shoulder of the testing robot is depicted in Figure 4.

Because the structure is divided into parts like the wrist, elbow, and shoulder, it may move in a manner akin to that of the human body. Joints connect these segments.

Figure 3. Testing robot arm base waist

The arm base waist of the testing robot is seen in Figure 3. The physical framework that establishes the shape and capabilities of a robotic manipulator is known as the robotic arm structure. This structure, which is made up of several interconnected parts, is intended to resemble the suppleness and agility of a human arm.

A robotic arm usually has several joints or links that can move in either a rotational or translational direction. It also frequently has end effectors for certain functions. Actuators—typically motor or servos—connect the joints and propel the arm's motion. The arm shoulder of the testing robot is depicted in Figure 4.

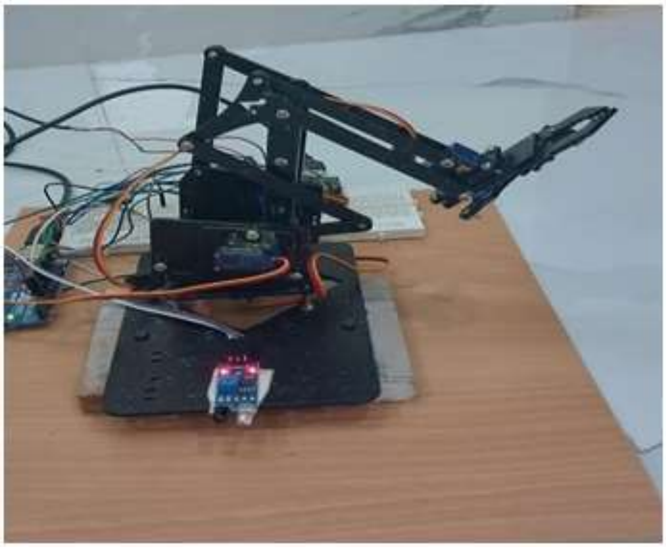

Figure 4. Testing robot arm shoulder

The construction is arranged into parts, such as the shoulder, elbow, and wrist, allowing for a range of motion akin to the human body. Joints connect these segments.

V. CONCLUSION

To sum up, the Arduino-based robotic arm control system is a major advancement in the combination of automation and widely available technology. In addition to making programming and control easier, the use of Arduino as the primary microcontroller has democratized robotics by making it accessible to both professionals and hobbyists. It is now evident from observation that robotic arm movement is precise, accurate, controllable, and easy to operate. The robotic arm has solved the earlier issues and can now pick up and position dangerous things with ease and accuracy.

REFERENCES

1. N. K. Agrawal, V. K. Singh, V. S. Parmar, V. K. Sharma, D. Singh, and M. Agrawal, "Design and development of IoT based robotic arm by using Arduino," in Proceedings of the 4th International Conference on Computing Methodologies and Communication, ICCMC 2020, Mar. 2020, pp. 776–780, doi: 10.1109/ICCMC48092.2020.ICCMC-000144.

2. S. H. Chua, J. S. Limqueco, E. L. Lu, S. W. Que, and D. Abuan, “Development of a microcontroller-based wireless writing robotic arm controlled by skeletal tracking,” Nov. 2018, doi: 10.1109/HNICEM.2018.8666384.
3. N. M. Sundaram, R. Shiva, R. Krishnan, S. Kamal, J. Rajeswari, and U. G. Students, “Robotic vehicle movement and arm control through hand gestures using Arduino,” International Research Journal of Engineering and Technology, vol. 05, no. 4, pp. 1189– 1194, 2018.
4. R. Rahman, M. Sajid Rahman, and J. R. Bhuiyan, “Joystick controlled industrial robotic system with robotic arm,” in 2019 IEEE International Conference on Robotics, Automation, Artificial-Intelligence and Internet-of-Things, RAAICON 2019, Nov. 2019, pp. 31–34, doi: 10.1109/RAAICON48939.2019.18.
5. S. C. Suhaimin, N. L. Azmi, M. M. Rahman, and H. M. Yusof, “Analysis of point-to-point robotic arm control using PID controller,” Oct. 2019, doi: 10.1109/ICOM47790.2019.8952036.
6. A.Sabarivani, A. Anbarasi, A. Vijayaiyyappan, and S. Sindhuja, “Wireless synchronization of robotic arm with human movements using aurdino for bomb disposal,” in Proceedings of the International Conference on Smart Systems and Inventive Technology, ICSSIT 2018, Dec. 2018, pp. 229–234, doi: 10.1109/ICSSIT.2018.8748690.
7. Z. Wang, D. Chen, and P. Xiao, “Design of a voice control 6DOF grasping robotic arm based on ultrasonic sensor, computer vision and alexa voice assistance,” in Proceedings - 10th International Conference on Information Technology in Medicine and Education, ITME 2019, Aug. 2019, pp. 649–654, doi: 10.1109/ITME.2019.00150.
8. P. Sihombing, M. B. Rifky, H. Herriyance, and E. Elviwani, “Robotic arm controlling based on fingers and hand gesture,” in MECnIT 2020 - International Conference on Mechanical, Electronics, Computer, and Industrial Technology, Jun. 2020, pp. 40– 45, doi: 10.1109/MECnIT48290.2020.9166592.
9. T. Vignesh, R. Nivetha, G. Mowneka, and D. Selvakumar, “Design and fabrication of 6-axis gesture controlled robot,” in 2019 5th International Conference on Advanced Computing and Communication Systems, ICACCS 2019, Mar. 2019, pp. 246–250, doi: 10.1109/ICACCS.2019.8728465.
10. L. M. V. Creo, G. M. Dacanay, L. C. P. Jarque, C. J. P. Umali, and E. R. E. Tolentino, “Controlling the bomb disposal robot using Microsoft Kinect sensor,” Jun. 2021, doi: 10.1109/ICCISc52257.2021.9485010.
11. S. S. Dheeban, D. V. Harish, A. Hari Vignesh, and M. Prasanna, “Arduino Controlled Gesture Robot,” Dec. 2018, doi: 10.1109/ROMA46407.2018.8986730.
12. R. M. Ismail, S. Muthukumaraswamy, and A. Sasikala, “Military support and rescue robot,” in Proceedings of the International Conference on Intelligent Computing and Control Systems, ICICCS 2020, May 2020, pp. 156–162, doi: 10.1109/ICICCS48265.2020.9121041.
13. A.B. Bakri, R. Adnan, and F. A. Ruslan, “Wireless hand gesture controlled robotic arm via NRF24L01 transceiver,” in ISCAIE 2019 - 2019 IEEE Symposium on Computer Applications and Industrial Electronics, Apr. 2019, pp. 16–22, doi: 10.1109/ISCAIE.2019.8743772.
14. E. B. Mathew, D. Khanduja, B. Sapra, and B. Bhushan, “Robotic arm control through human arm movement detection using potentiometers,” in 2015 International Conference on Recent Developments in Control, Automation and Power Engineering, RDCAPE 2015, Mar. 2015, pp. 298–303, doi: 10.1109/RDCAPE.2015.7281413.
15. L. D. William Raj, R. Abinaya, A. Brundha, N. D. Lakshmi, and S. Geethapriya, “Robotic Arm for extricate operation in bore well,” Jul. 2020, doi: 10.1109/ICSCAN49426.2020.9262297.
16. W. M. H. W. Kadir, R. E. Samin, and B. S. K. Ibrahim, “Internet controlled robotic arm,” Procedia Engineering, vol. 41, pp. 1065–1071, 2012, doi: 10.1016/j.proeng.2012.07.284.
17. R. K. Megalingam, S. Boddupalli, and K. G. S. Apuroop, “Robotic arm control through mimicking of miniature robotic arm,” in 2017 4th International Conference on Advanced Computing and Communication Systems, ICACCS 2017, Jan. 2017, doi: 10.1109/ICACCS.2017.8014622.
18. N. R. Babu, S. S. Sai, U. Kumar, K. Chenchireddy, K. T. Kumar, and M. Naveen, “Optimized speed control of BLDC motor control by using PID and ANFIS,” in International Conference on Sustainable Communication Networks and Application, ICSCNA 2023 - Proceedings, Nov. 2023, pp. 653–658, doi: 10.1109/ICSCNA58489.2023.10370045.
19. C. Guo and W. Luk, “FPGA-accelerated sim-to-real control policy learning for robotic arms,” IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 71, no. 3, pp. 1690–1694, Mar. 2024, doi: 10.1109/TCSII.2024.3353690.
20. R. Bouzid, J. Narayan, and H. Gritli, “Investigating neural network hyper parameter variations in robotic arm inverse kinematics for different arm lengths,” in 2024 3rd International Conference on Power, Control and Computing Technologies, ICPC2T 2024, Jan. 2024, pp. 351–356, doi: 10.1109/ICPC2T60072.2024.10474947.
21. K. Suphalak, N. Klanpet, N. Sikaressakul, and S. Prongnuch, “Robot arm control system via ethernet with kinect V2 camera for use in hazardous areas,” in 2024 1st International Conference on Robotics, Engineering, Science, and Technology, RESTCON 2024, Feb. 2024, pp. 175–180, doi: 10.1109/RESTCON60981.2024.10463582.
22. Z. Lu et al., “A wearable human-machine interactive instrument for controlling a wheelchair robotic arm

system," IEEE Transactions on Instrumentation and Measurement, vol. 73, pp. 1–15, 2024, doi: 10.1109/TIM.2024.3376685.

23. M. A. Khalifa, S. N. Mekid, and Z. H. Khan, "Position and torque joint control for precision drilling in robotic manufacturing applications," in 2024 ASU International Conference in Emerging Technologies for Sustainability and Intelligent Systems, ICETSIS 2024, Jan. 2024, pp. 888–893, doi: 10.1109/ICETSIS61505.2024.10459489