Direct Teaching Arm Robot System

Description

This project presents a direct teaching–based robotic arm system where the arm robot can be manually guided, and its movements can be recorded and replayed without requiring programming knowledge. The system addresses a common problem faced by users who need to operate a robotic arm for different tasks but do not have a background in programming. This approach is important because traditional robotic arm systems typically require coding and technical expertise, which limits accessibility for non-technical users. By enabling movement recording through direct teaching, this system makes robotic automation more intuitive, flexible, and user-friendly.

Objectives

The objectives of this project are To develop a robotic arm system that can be operated without writing program code, To allow users to teach movements directly by guiding the arm, To replay recorded movements accurately for task execution. Technically, the project focuses on capturing and storing coordinated joint positions from Dynamixel actuators during the teaching phase.

Technologies Used

- Dynamixel Actuators
- Arduino Uno
- Raspberry Pi
- Arduino IDE
- 3D Printing
- Autodesk Inventor
- Measuring Tools
- PLA Filament

System Workflow

1. Arm Robot Design and Fabrication The robotic arm was designed using Autodesk Inventor. Each mechanical component was then fabricated using a 3D printer based on the finalized design.

2. Component Assembly All printed parts, Dynamixel actuators, and electronic components were assembled according to the system design. Electrical connections between actuators, controllers, and power sources were carefully configured.
3. Program Development A direct teaching program was developed using Arduino IDE. The program enables users to record multiple motion points by pressing a button. The recorded sequence includes Starting point coordinate, Second point coordinate, Third point coordinate, Gripper close action,End point coordinate, Gripper open action, and Return to starting point. Each coordinate and gripper state is stored sequentially and executed during playback.
4. Program Testing The system was tested by recording various task motions and replaying them using the playback button. The accuracy, smoothness, and repeatability of the arm movement were evaluated.

Key Features

- Robotic arm operation without programming
- Direct teaching through manual guidance
- Recording of joint coordinates at each motion point
- Playback of recorded movements for task repetition

Challenges & Issues (Key Section)

- Arm movement was not smooth
- Incorrect dimensions in 3D-printed parts, especially holes due to printing tolerances
- Difficulty in assembling bolts and joints
- Gripper lacked sufficient gripping force

Solutions & Technical Decisions

- Reduced system delays to improve movement smoothness
- Reprinted components or manually drilled holes to achieve proper fit
- Used better tools to ease mechanical assembly
- Applied rubber material (3M tape) to the gripper surface to increase friction

Results & Evaluation

The implemented solutions successfully resolved most of the system issues, resulting in smoother arm movement and improved gripping performance. However, manual drilling of holes proved inefficient, indicating that future improvements should focus on refining the 3D design to reduce dimensional errors.

Lessons Learned

- Practical experience using Dynamixel actuators
- End-to-end process of 3D printing robotic components
- Proper 3D printer configuration and calibration
- Correct filament storage to maintain print quality
- Programming and system control using Arduino IDE