

eYSIP2017

ROBOTIC ARM



Aditya Gaddipati
Arjun Sadananda
Simranjeet, Lohit

Duration of Internship: 22/05/2017-07/07/2017

2017, e-Yantra Publication

Robotic Arm

Abstract

The motivation behind this project was to **DESIGN**(Mechanical) a robotic arm and **PLAN**(Computer Science) and **CONTROL**(Electronics) the motion of the arm in real time using **Kinect- Point Cloud**.

Completion status

Give details for work/project completed successfully. If work is not complete, mention the details till which task is done.

- **Designed a Robotic Arm** with *6 DOF powered by Dynamixel AX12A*, resembling the dynamics of a *real human arm* from shoulder, elbow to wrist. Complete with a gripper design powered by MicroServo(9g).
- **Controlled** the arm using Arduino Mega (Atmega 2560) through a **GUI** created using Processing.
- **Interfaced** 6 Dynamixel AX 12A motors with ROS, using Arduino Mega to control the **flow of packets** between **dynamixel_motor** package in ROS and Dynamixel.
- Created a **URDF model** of the arm and mirrored(visualized) the real model movements on the mathematical model using the feedback from Dynamixel on **RViz**.
- Built a **PCB(Arduino Sheild)** to interface Arduino Mega with Dynamixel via *74LS241 buffer* and power the arm from SMP.
- Used **MoveIt** to generate a trajectory for the robotic arm to move to goal position and send the instructions to the arm.
Not Reliable yet... requires more tests and corrections for reliable motion execution.
- Used **PointCloudLibrary** on roscpp to process the point cloud from the **Kinect** to get clusters of objects on a table.
Object recognition is not achieved.



1.1. HARDWARE PARTS

- ### 1.1 Hardware parts
- 6 Dynamixel AX 12A [Datasheet](#),
 - Arduino Mega (Atmega 2560) [Datasheet](#)
 - 1 SMP
 - 1 74LS241 buffer [Datasheet](#)
 - Connection diagram

1.2 Software used

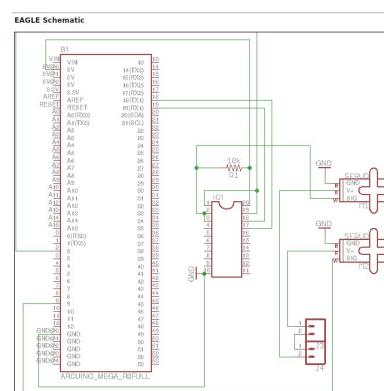
- Fusion 360: version [Official Website](#)
- Cura ver 2.5.0 [Official Website](#)
- ROS Indigo [Official Website](#)
Packages:
 - dynamixel_motor
 - MoveIt
 - freenect_launch
- Arduino IDE [Official Website](#)
- Processing IDE [Official Website](#)
- Library: [PointCloudLibrary](#)

1.3 Assembly of hardware

Circuit diagram and Steps of assembly of hardware with pictures for each step

Circuit Diagram

Circuit schematic from Eagle



Arm Assembly

Assembling Robotic Arm is self explanatory.

1.4. SOFTWARE AND CODE

Parts

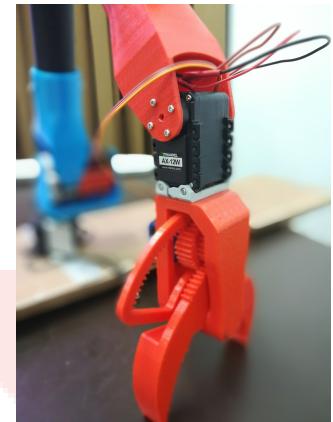
Shoulder



Elbow



Wrist and Grabber



1.4 Software and Code

[Github link](#) for the repository of code

- dxl_pc_bridge.ino: Arduino Code for controlling flow of packets.
- dxl_controller.ino: Arduino code for controlling the robotic arm.
- dxl_controller.pde: Processing code for the processing GUI.
- control_manager.launch: Starts Controller_Manager
- robot_state.launch: Publishes current state of the robot.
- start_meta_controller.launch: Starts joint trajectory action controller.
- robotic_arm_bringup_rviz.launch: Starts MoveIt!
- Object Detection Directory: roscode to subscribe to kinect raw point cloud and publishes processed point cloud
 - VoxelGrid Downsampling
 - RANSAC Plane Detection
 - Conditional Removal
 - Euclidean Clustering



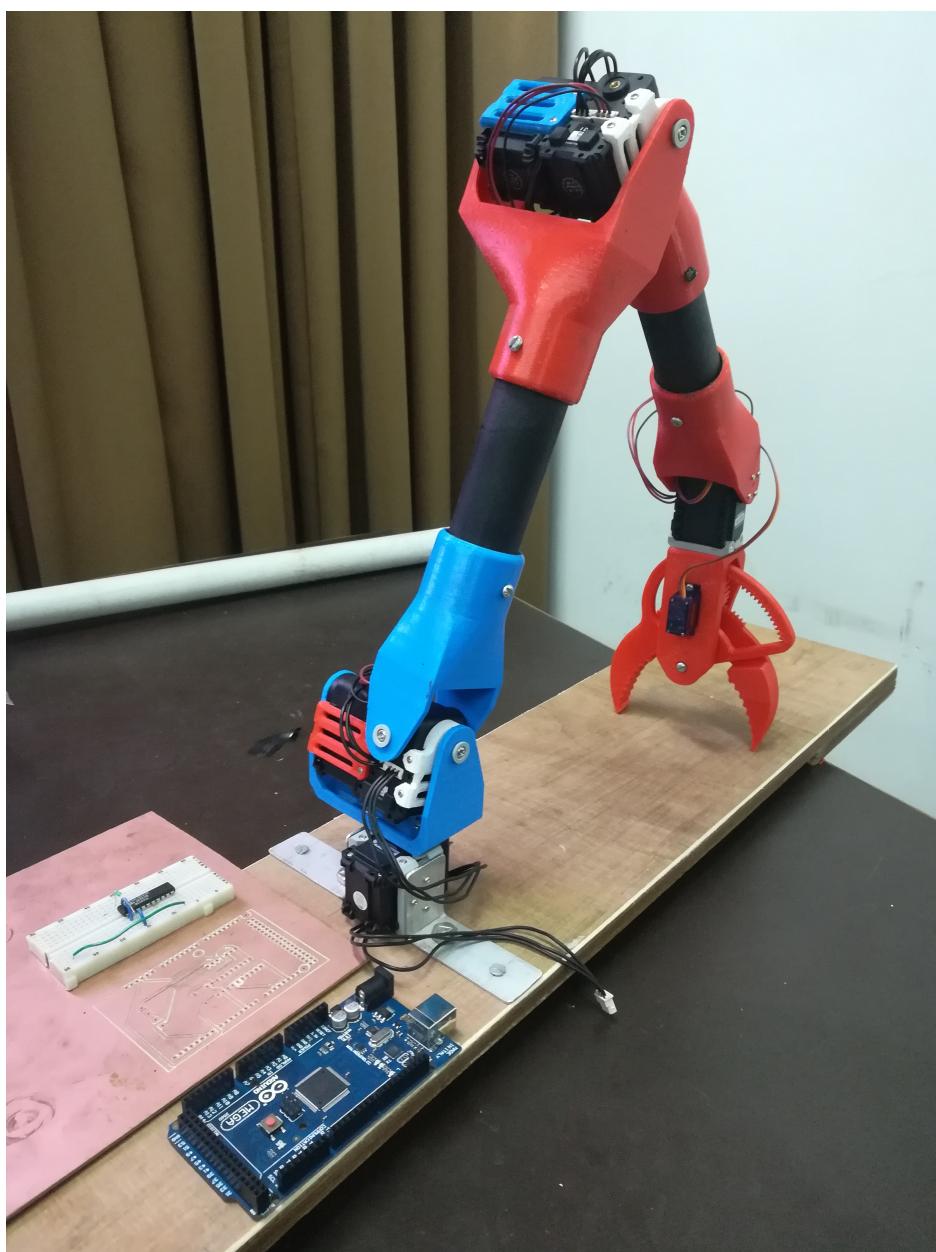
1.5. USE AND DEMO

1.5 Use and Demo

Demo

[Youtube Link](#) of demonstration video

Full Assembly





1.6. FUTURE WORK

- ### 1.6 Future Work
- MoveIt: Execution of MoveIt must be tested and necessary corrections must be done to achieve consistent results.
 - Kinect: Continue work with pcl. Use the cluster along with color information and feed it to machine learning module to achieve object recognition.
 - Design: Modification to design can be done to achieve greater range of motion. Ex: slight change in elbow design can increase downward motion of the fore arm.

1.7 Bug report and Challenges

- sync_write function in dynamixel_motor -> dynamixel_driver -> dynamixel_io.py needed to be modified to call write for every ID, Since in Arduino Code - dxl_serial we couldnt handle broadcast packages.
- Dynamixel AX 12A ID=3 was replaced with ID=6 and ID=6 with AX12W since it completely stopped responding and AX12A ID=1 - ShoulderRoll stopped giving position feedback. These two issues need to be looked into. Temporarily we replaced these with AX12W motors.
- The present design provides 10 kg.cm load at worst cases, but for smooth movement load must be kept below 1/5th of max load i.e. $1/5 \times 15\text{kg.cm}$

Bibliography

- [1] Mastering ROS for Robotics Programming: by Lentin Joseph
- [2] ROS by Example Vol2: by R. Patrick Goebel
- [3] MoveIt: <http://moveit.ros.org/>
- [4] ROS: http://wiki.ros.org/dynamixel_motor
- [5] Fusion 360: Youtube Tutorials by Channel: Autodesk Fusion 360
- [6] thingiverse.com: *WonderTiger, Gripper Design*