Hazardous Area Robotic Arm (HARA)

Erin Morton, Rob Moran, Craig Rich & Rj Woodward
Advisor: Bo Tao
School of Engineering

Abstract

- Robotic arm replicates movements made by the user wearing the control sleeve.
- The robot may be used by essential employees to maintain a safe distance from possible exposures.

- Measurable Objectives:
  - Payload of 0.25 lbs.
  - Reaction time of 0.1 seconds
  - Radius of motion to be 1 foot

Introduction

As a group, we wanted to make something that would help benefit those who are most in danger of contracting COVID-19. HARA was initially designed to assist hospital workers checking up on patients, allowing them to help give care from a safe distance. HARA's design has since evolved to be able to assist any essential worker that comes in frequent contact with the public and germs. This includes store cashiers, toll booth workers and much more.

Project Plan

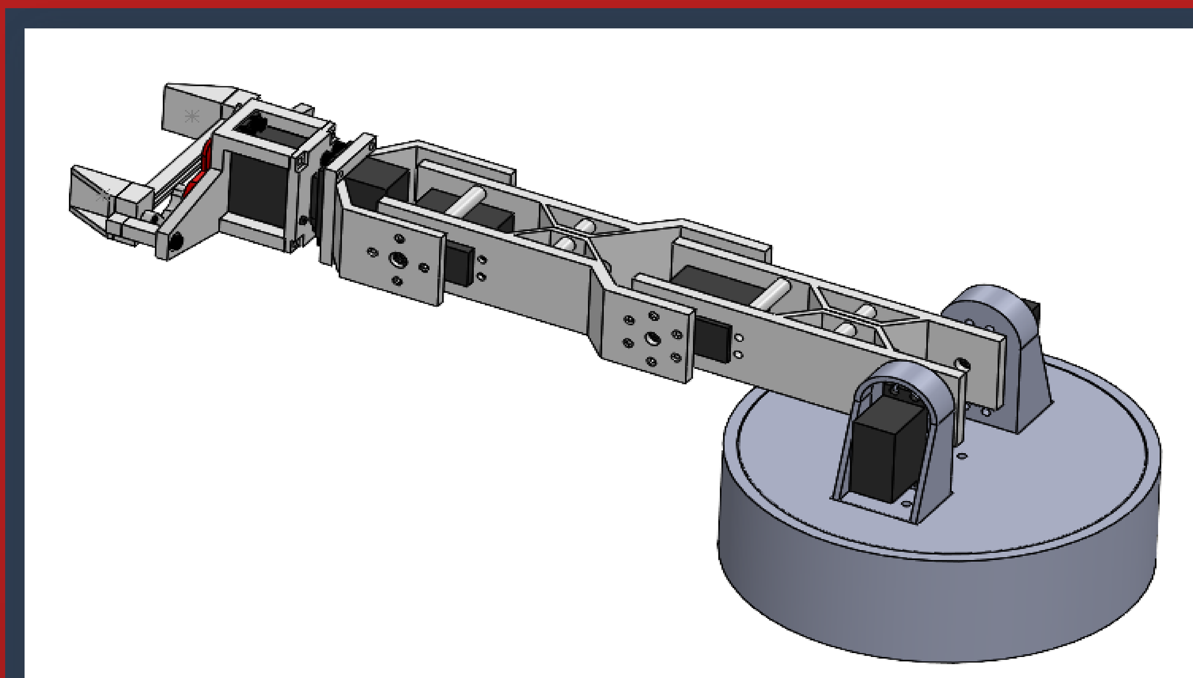
1. Brainstorm/Research
2. Create team roles and divide tasks
   1. Create CAD model
   2. Create Arduino code
   3. Determine proper loading specifications
   4. Build Arduino schematics
3. Assemble
4. Testing
5. Improvements and final product

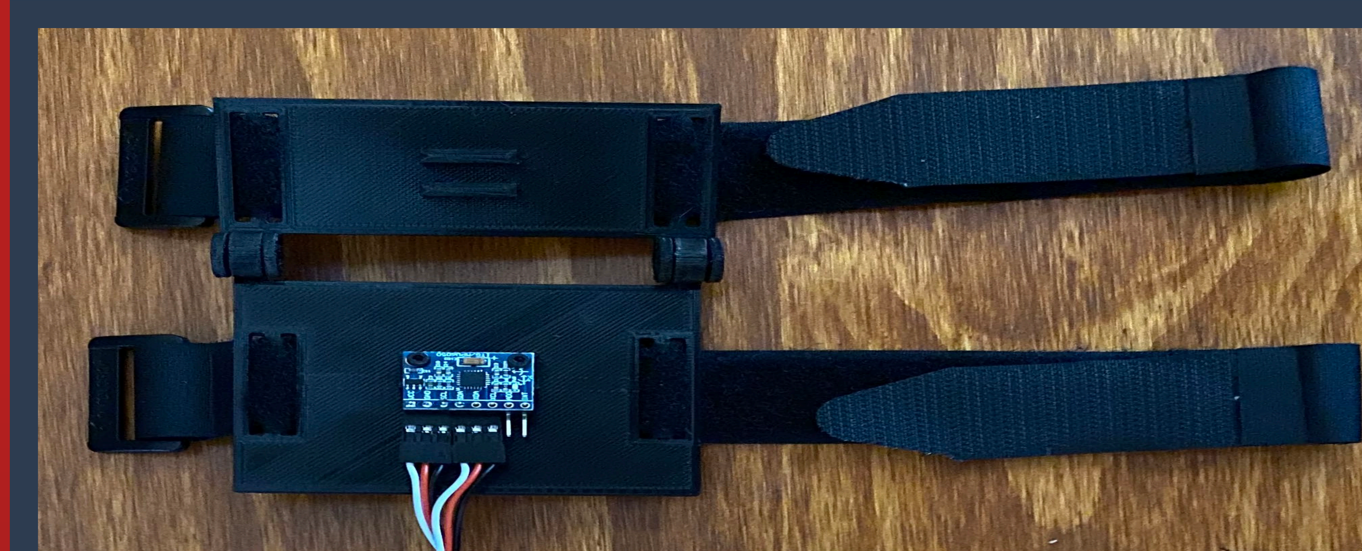
Methodology

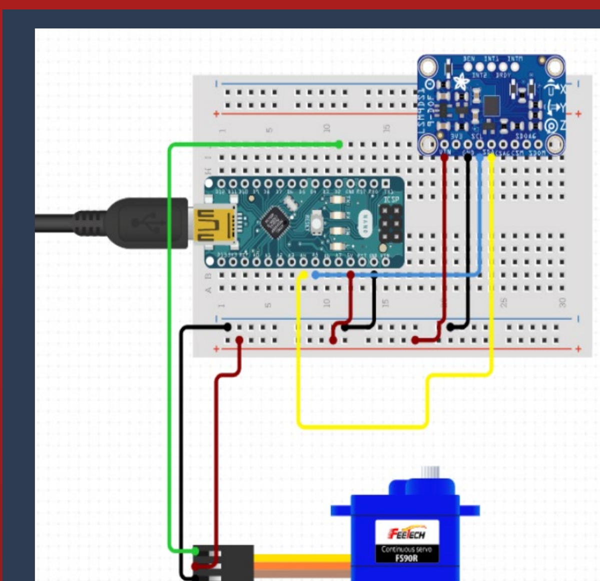
Placement of sensors on arm:
- Claw flex sensor is placed along the fingers
- Wrist sensor is placed on top of the hand. This will measure wrist bending and rotation.
- Elbow sensor is placed along the back of the forearm.
- The shoulder sensor is placed along the back of the upper arm. This measures all axis' and will manipulate the base component of HARA.

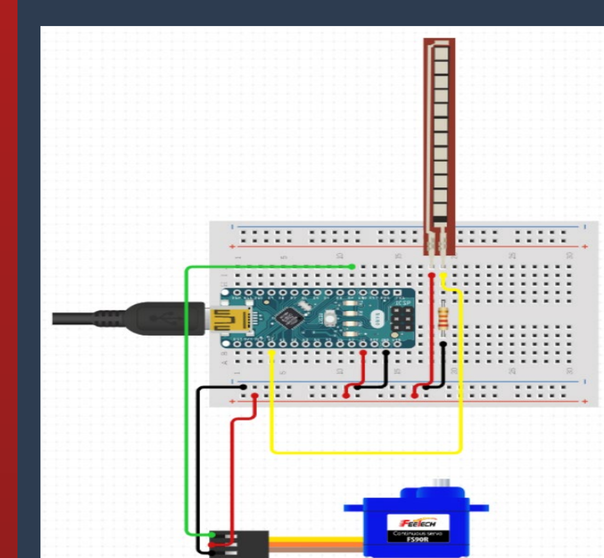
Servo Selection Process
- Based on Factor of Safety (2),
- Must conform with power (5V) and joint rotation requirements
- Servo must meet or exceed torque values found in the Calculations Section
- Unit weight must be deemed appropriate versus torque supplied

Models, Assemblies, & Schematics









Reference

[1] Mechanical Engineering Department. "MECH3000 Design of Machine Elements". Lab Handout. Wentworth Institute of Technology
[2] "MPU 6050 Tutorial: How to Program Mpu 6050 with Arduino." *Arduino Project Hub*, create.arduino.cc/projecthub/MissionCritical/mpu-6050-tutorial-how-to-program-mpu-6050-with-arduino-aee39a.
[3] Zoysa, Kavindu Gimhan. "Lets Work with MPU6050 (GY-521) - Part1." *Medium*, Medium, 26 Dec. 2017, medium.com/@kavindugimhanzoysa/lets-work-with-mpu6050-gy-521-part1-6db0d47a35e6.

Calculations

Wrist, elbow, and shoulder calculations are shown below for information used to determine the torque required for each servo.

Wrist Calculations					
Part	Distance from CoM (mm)	*	Mass (g)	=	Resultant Torque (g•mm)
Weight	137.725	*	113.500	=	15,631.788
HARA-X-002	94.667	*	111.970	=	10,599.864
HARA-X-003	32.137	*	88.730	=	2,851.516
			Total Torque		29,083.168

To convert g•mm to kg•cm, divide by 10,000	Required Torque on Joint: 2.908 kg•cm
Targeted Factor of Satey: 2	Targeted Torque With FOS of 2: 5.816 kg•cm

Elbow Calculations					
Part	Distance from CoM (mm)	*	Mass (g)	=	Resultant Torque (g•mm)
Weight	265.175	*	113.500	=	30,097.363
HARA-X-002	224.082	*	111.970	=	25,090.462
HARA-X-003	160.000	*	88.730	=	14,196.800
HARA-X-004	83.888	*	140.680	=	11,801.364
			Total Torque		81,185.988

To convert g•mm to kg•cm, divide by 10,000	Required Torque on Joint: 8.119 kg•cm
Targeted Factor of Satey: 2	Targeted Torque With FOS of 2: 16.238 kg•cm

Shoulder Calculations					
Part	Distance from CoM (mm)	*	Mass (g)	=	Resultant Torque (g•mm)
Weight	389.625	*	113.500	=	44,222.438
HARA-X-002	349.530	*	111.970	=	39,136.874
HARA-X-003	286.030	*	88.730	=	25,379.442
HARA-X-004	208.000	*	153.680	=	31,965.440
HARA-X-005	84.000	*	158.880	=	13,345.920
			Total Torque		154,050.114

To convert g•mm to kg•cm, divide by 10,000	Required Torque on Joint: 15.405 kg•cm
Targeted Factor of Satey: 2	Targeted Torque With FOS of 2: 30.81 kg•cm

Special thanks to Professor Michael Cameron and Professor Bo Tao for their contributions to this project.