Parallel Kinematic Manipulator (PKM) MQP 2011-2012

Introduction – PKM

M. Spenser Brouwer
Mechanical
Engineering ‘12

John W. Cushion
Mechanical
Engineering ‘12

Joshua Janssen
Mechanical
Engineering ‘12

Sean Townsend
Mechanical
Engineering ‘12

Elizabeth De Zulueta
Robotics
Engineering ‘12

Advisor:
Professor Stephen Nestinger,
ME/RBE Department

Purpose – PKM

Design, fabricate and implement a PKM robot with open architecture to be used in the Industrial Robotics curriculum for inverse kinematics and other classroom projects.

Background- PKM

Serial Manipulator

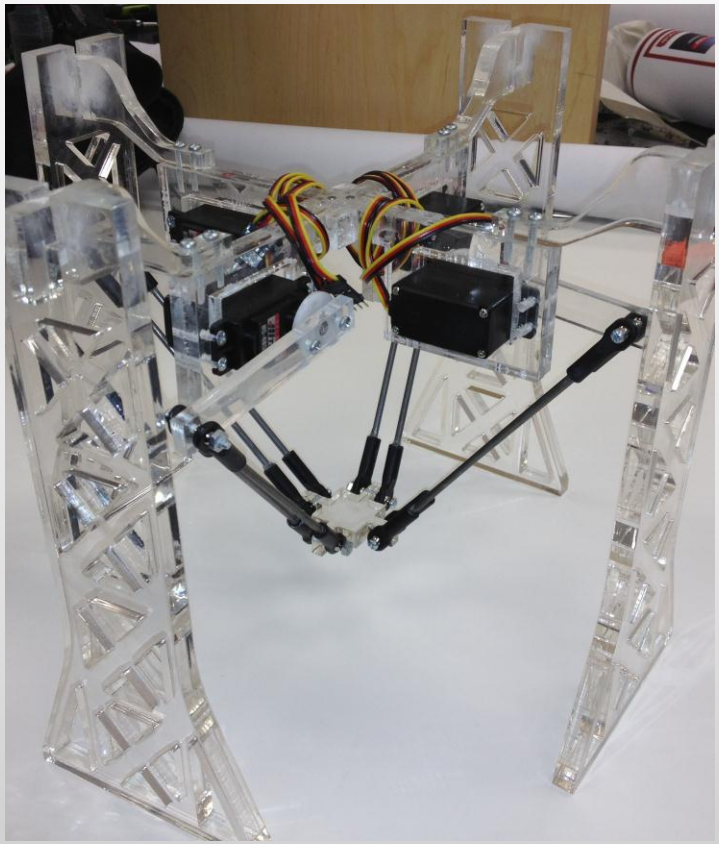
Parallel Kinematic Manipulator

Methodology –PKM

- Preliminary design research and analysis
- Prototyping for proof of concept
- Kinematics and initial design
- Part modeling and materials selection
- Fabrication, assembly and coding

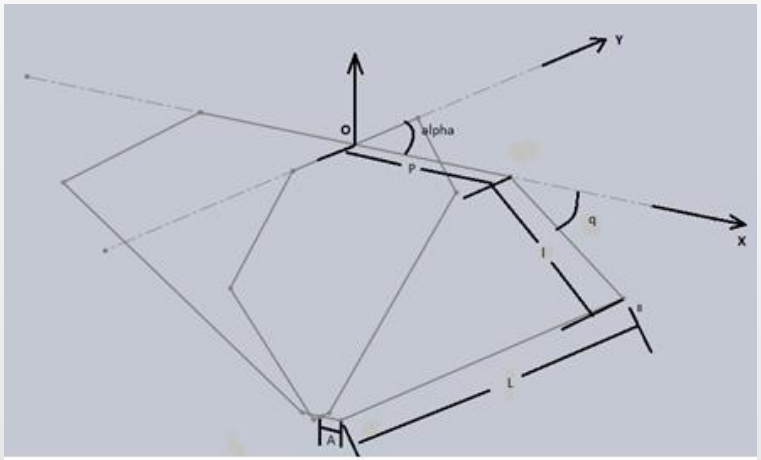
Mechanical

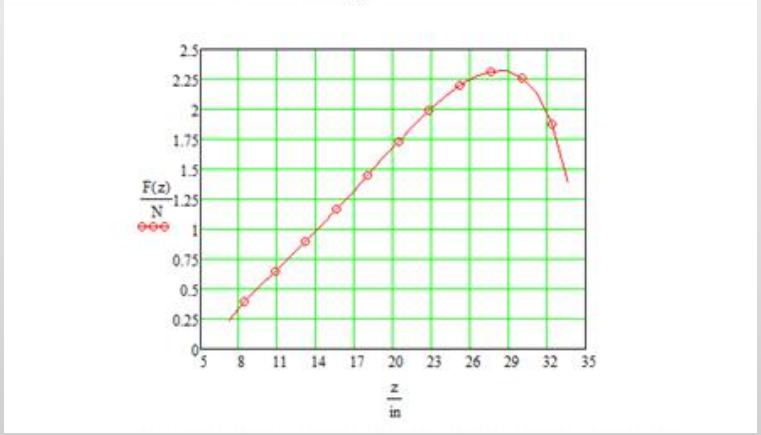
Prototyping



- Selection Process
- Delta
- Quattro

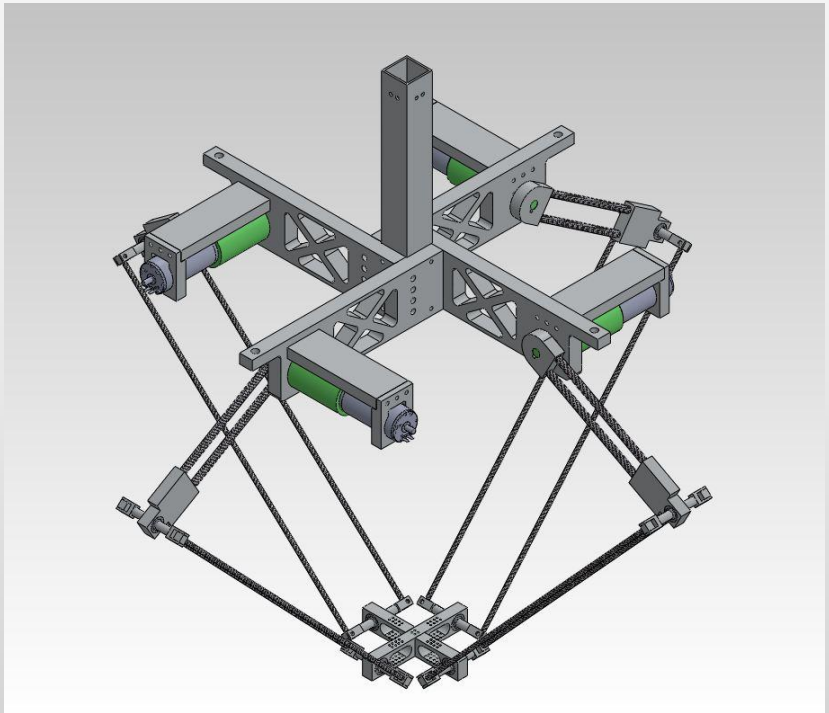
Analysis





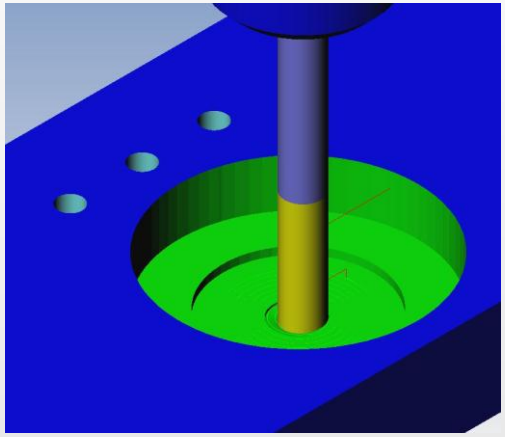
- Inverse Kinematics
- System parameter selection
- 100g Test
- Motor Selection

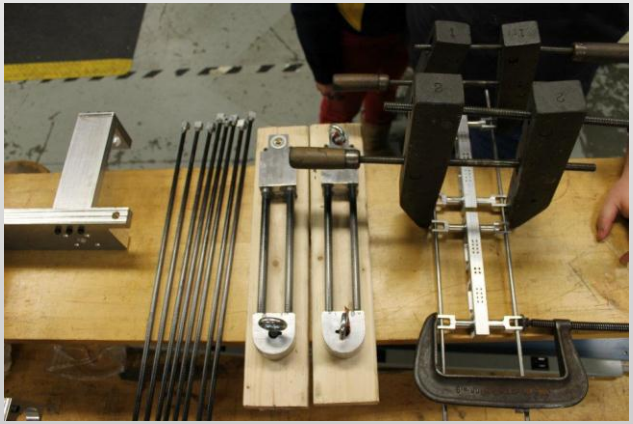
Design



- Kinematic Model for Analysis
- Overcoming Challenges

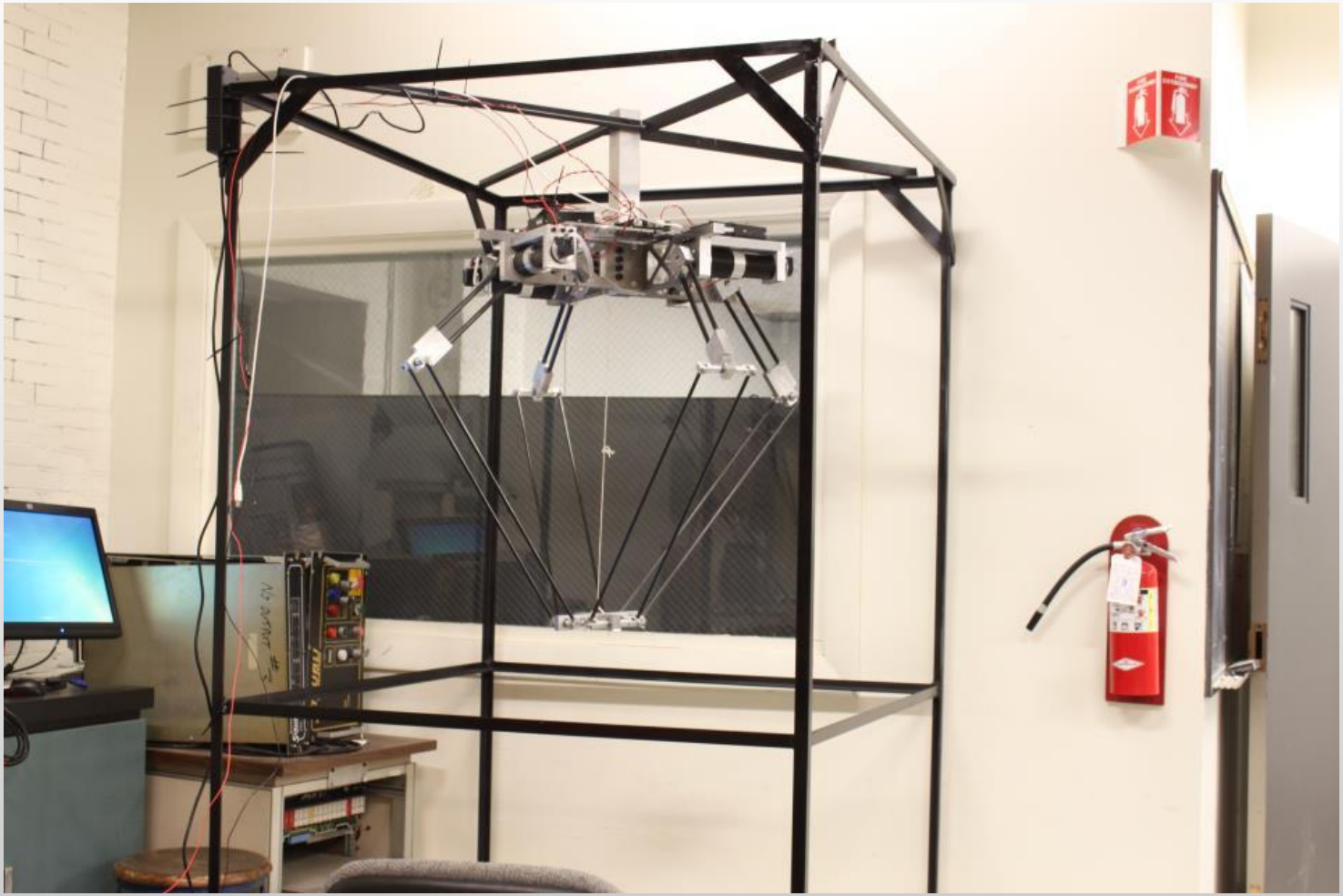
Manufacturing and Assembly





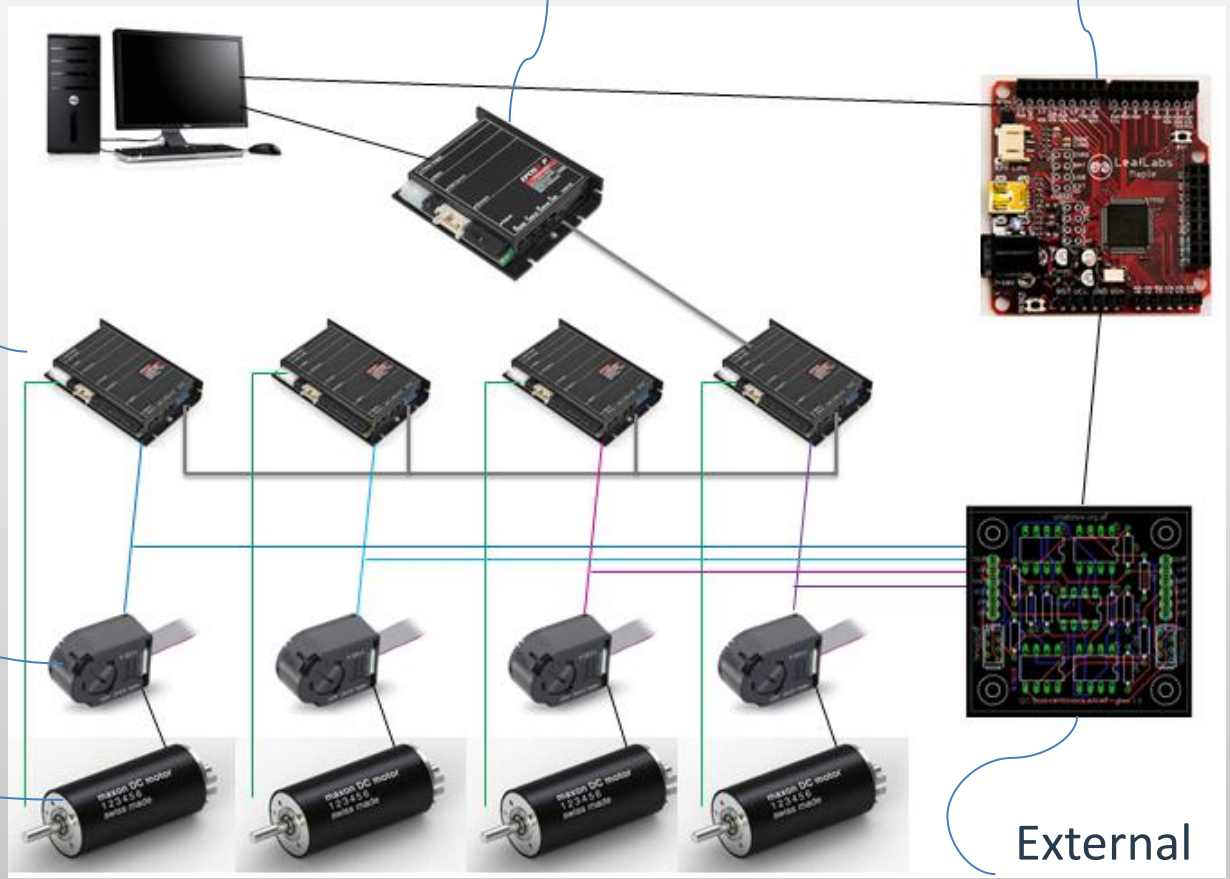
- Simulation
- 54 Custom Machined Parts
- Assembly Process
  - Jigs
  - Adhesive Choice
  - Errors

Final Product

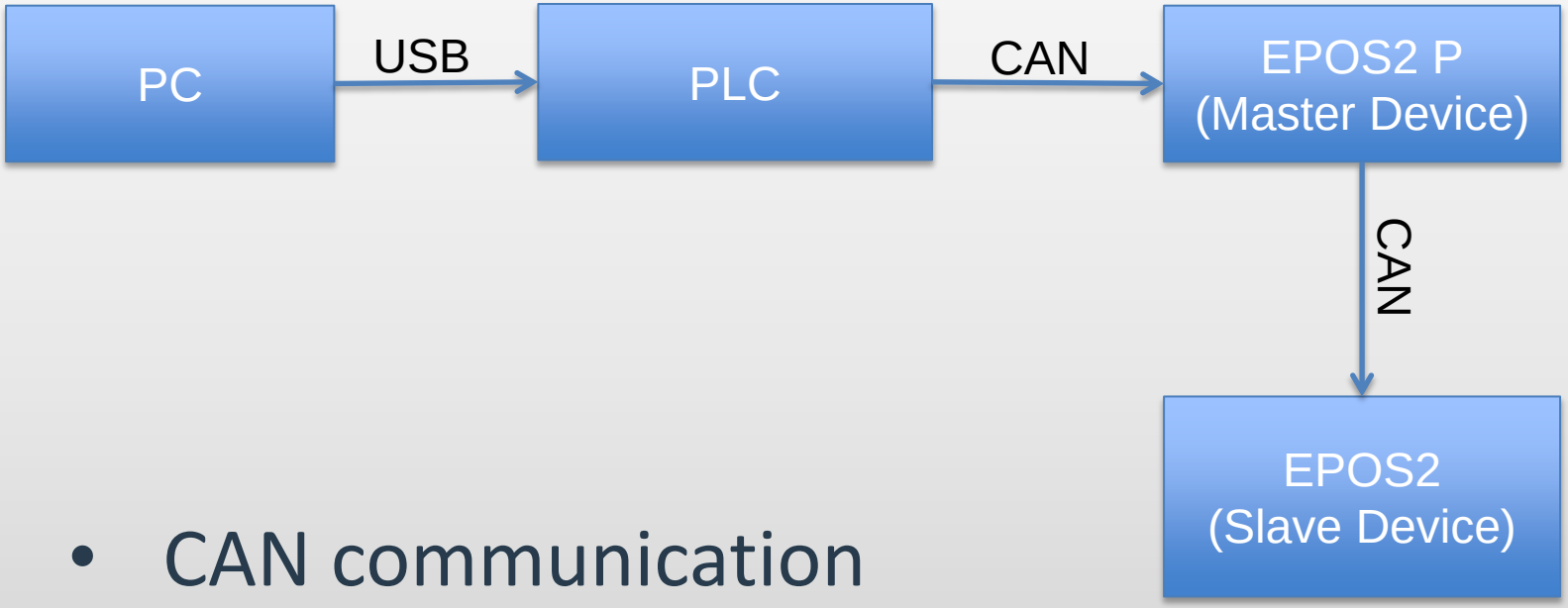


Electrical

Electrical Design – PKM

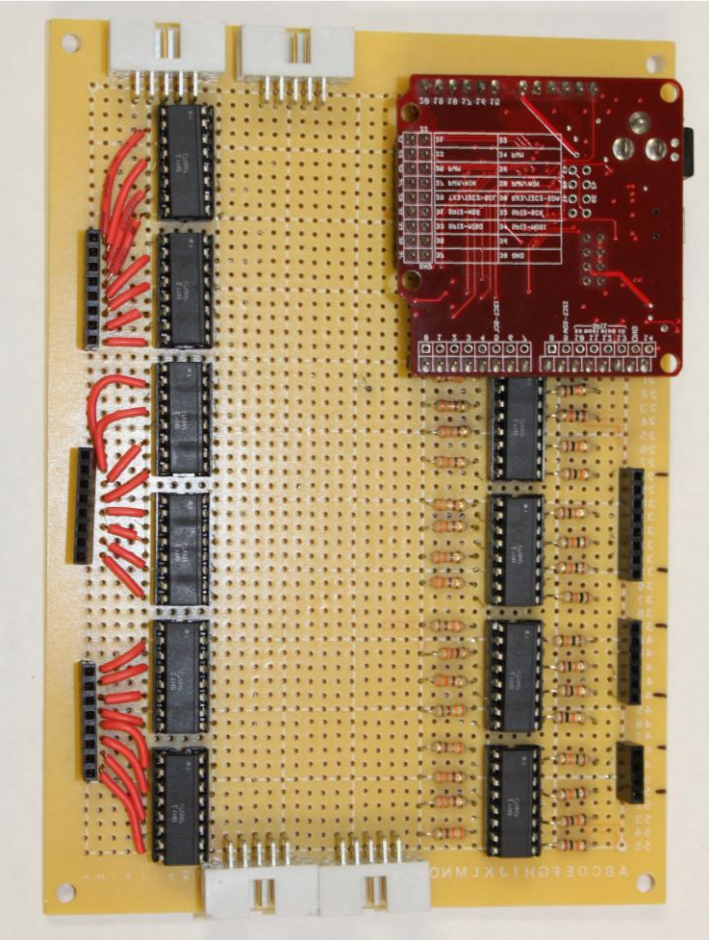


Programmable Logic Controller (PLC)



- CAN communication
- PLC to synchronize the motion control

Interface Board – PKM



- Open Architecture
  - Interaction with the robot at a lower level
- Input Signals
  - Analog and PWM
- Output Signals
  - Digital (Encoder feedback)
- Maple Code which takes a PWM signal
- Shield electronics using opto-couplers
  - Non-linearity
  - Smaller pull-up resister

Programming

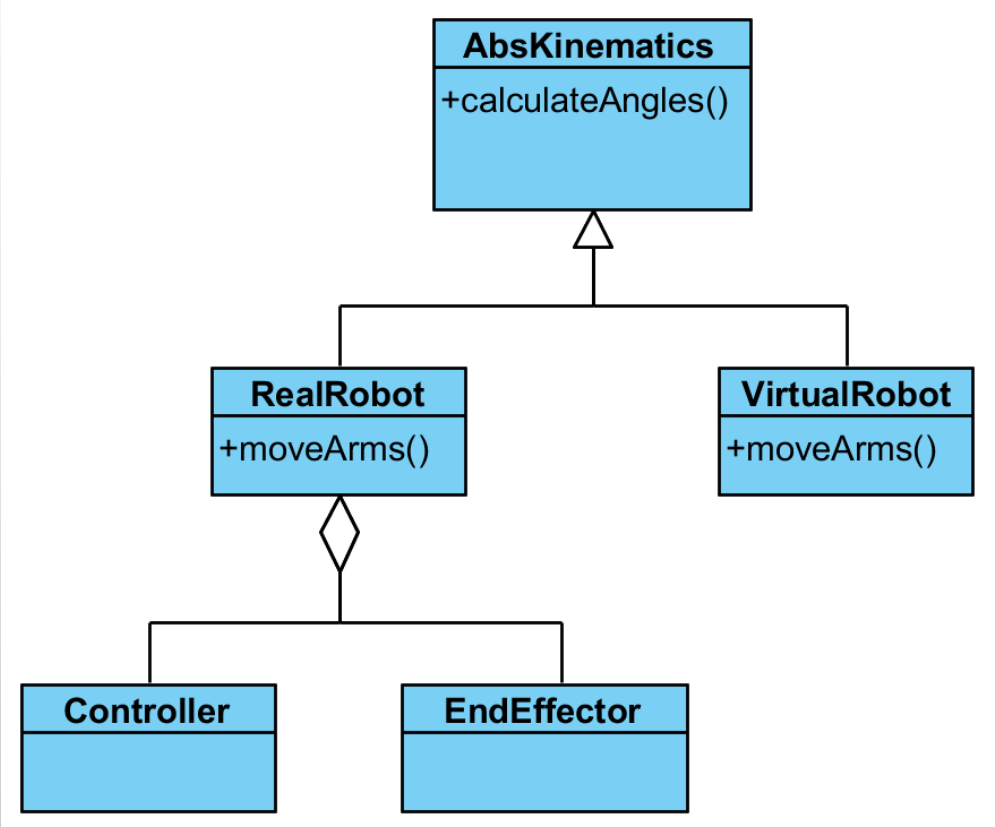
Programming Methodology

- Prototype coding for proof of concept
- Use cases and UML Diagrams
- Coding
  - Primarily in Java
  - EPOS libraries are in C++
  - Wrappers between languages

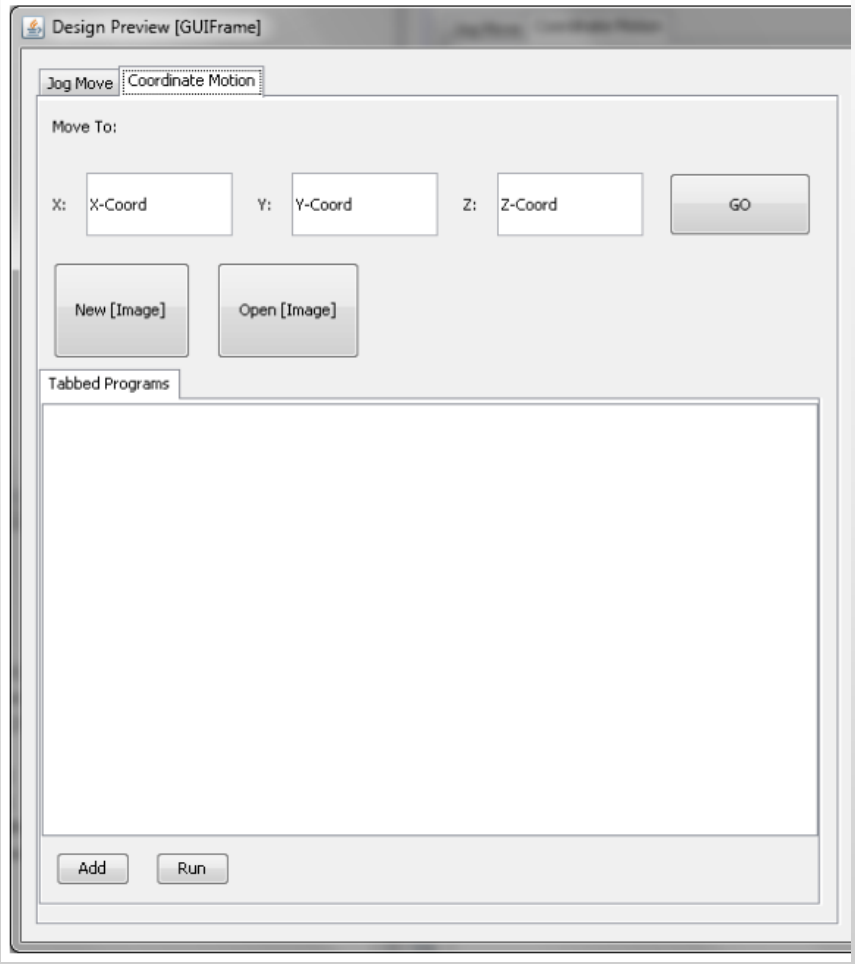
Use Cases and UML Diagrams

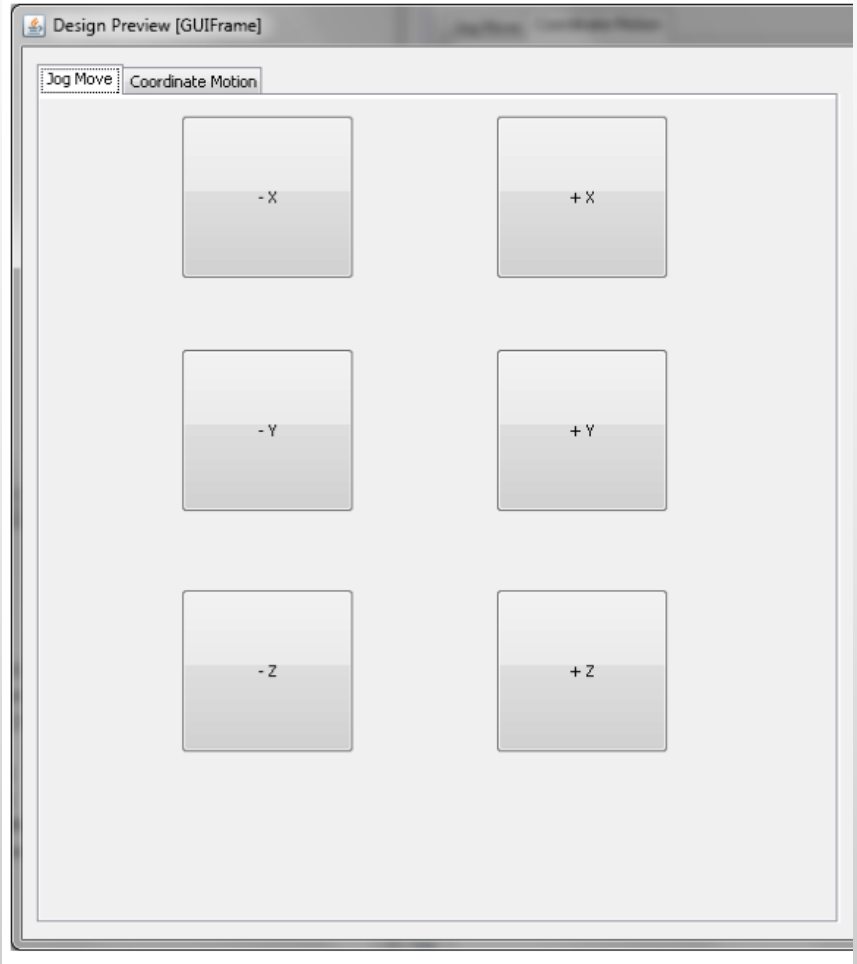
Use Cases:

- Jog Move
- XYZ Move
- Speed Control
- Origin Offset



GUI


Design Preview [GUIFrame]
Jog Move
Coordinate Motion
Move To:
X:
X-Coord
Y:
Y-Coord
Z:
Z-Coord
GO
New [Image]
Open [Image]
Tabbed Programs
Add
Run


Design Preview [GUIFrame]
Jog Move
Coordinate Motion
- X
+ X
- Y
+ Y
- Z
+ Z

Parallel Kinematic Manipulator (PKM) MQP 2011-2012

Questions?