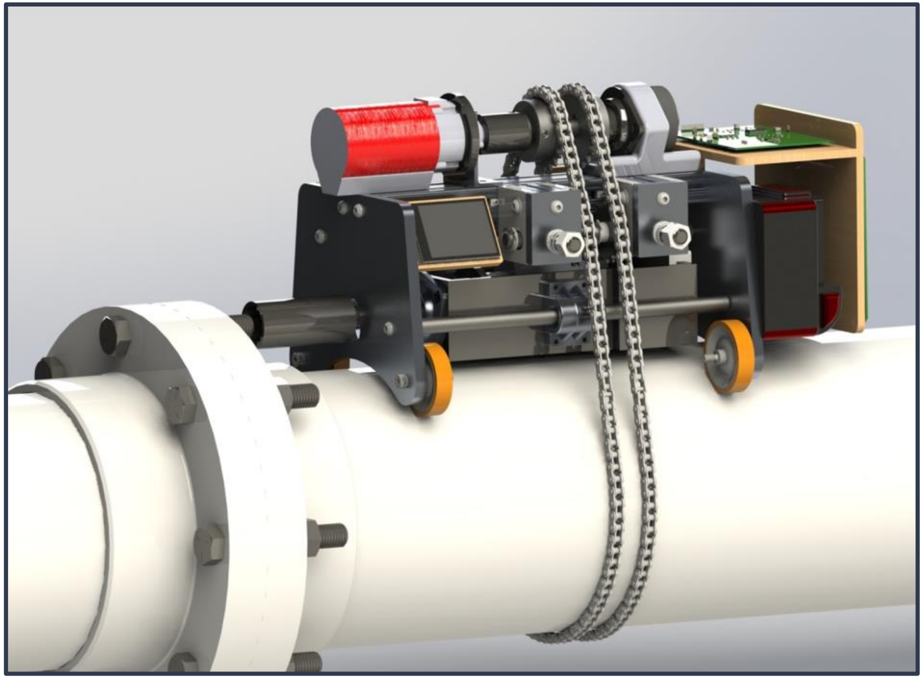
UCF COBOT – Robotic Flange Assembly

Senior Design 2018-2019

A Multidisciplinary Effort



M.E. Team
Fernando Gil
Juan Meneses
Justin Connolly
Reed Snowden
Rodrigo Duran
Juan Barajas

E.C.E. Team
Alana Icenroad
Cassidy Lyons
Tony Buda
Viviana Gonzalez

C.S. Team
Deepak Gunturu
Sopheap Chea
Tyler Teixeira

I.E. Team
Sara Alvarado
Yannick Giraud
Keaton Marn
Allison Randall

UCF

Automated Flange Assembly

- Manual flange assembly is a fatiguing & tedious process
- Siemens wants to improve this process
- What is a Cobot?
- Our end goal for the Cobot

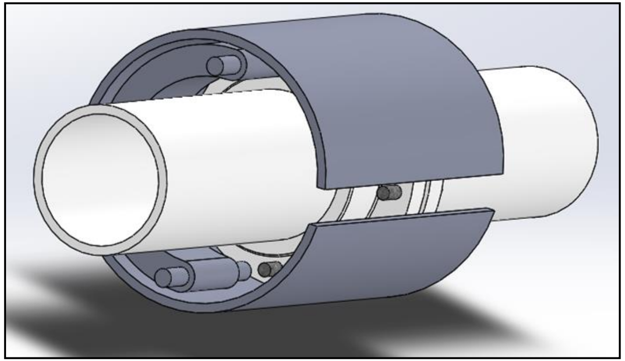
Outline

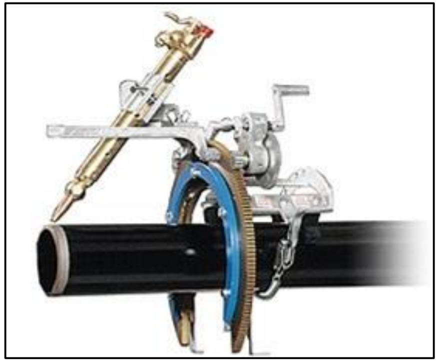
- Evolution of Cobot
- What we did
  - M.E. - Breakdown of subassembly systems
  - E.C.E - Electrical Components
  - C.S. - Simulation demo
- What each team learned
- Demonstration

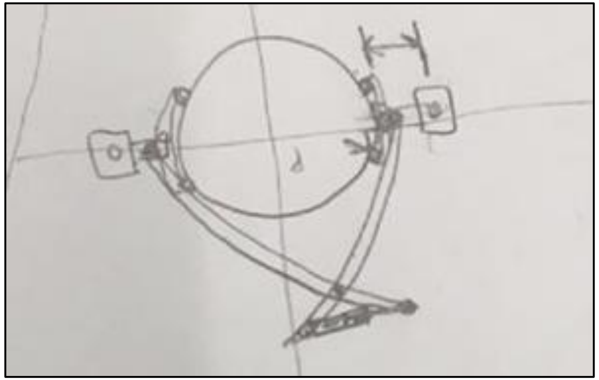
Why the Cobot

Considerations:

- Budget
- Manufacturability
- Adaptability
- Weight
- Installation Time
- Portability
- Required worker input

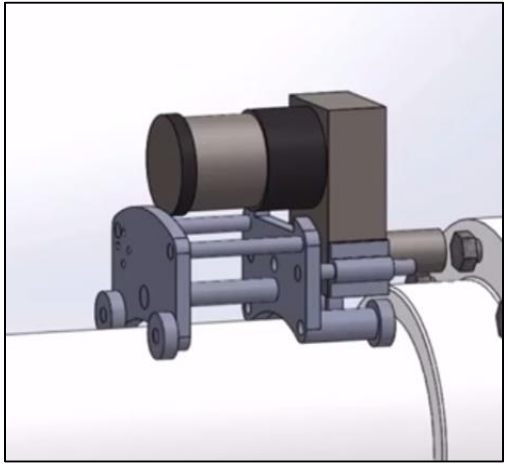




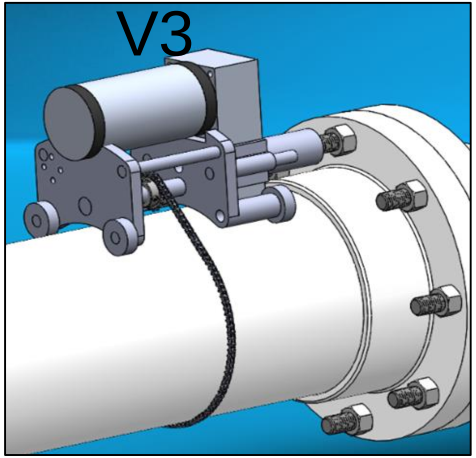


Cobot Evolution

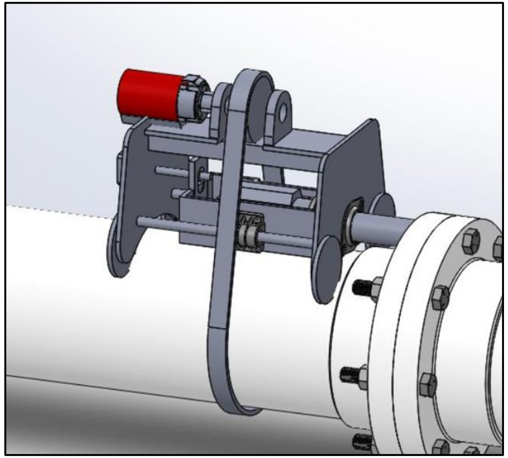
V1



V3



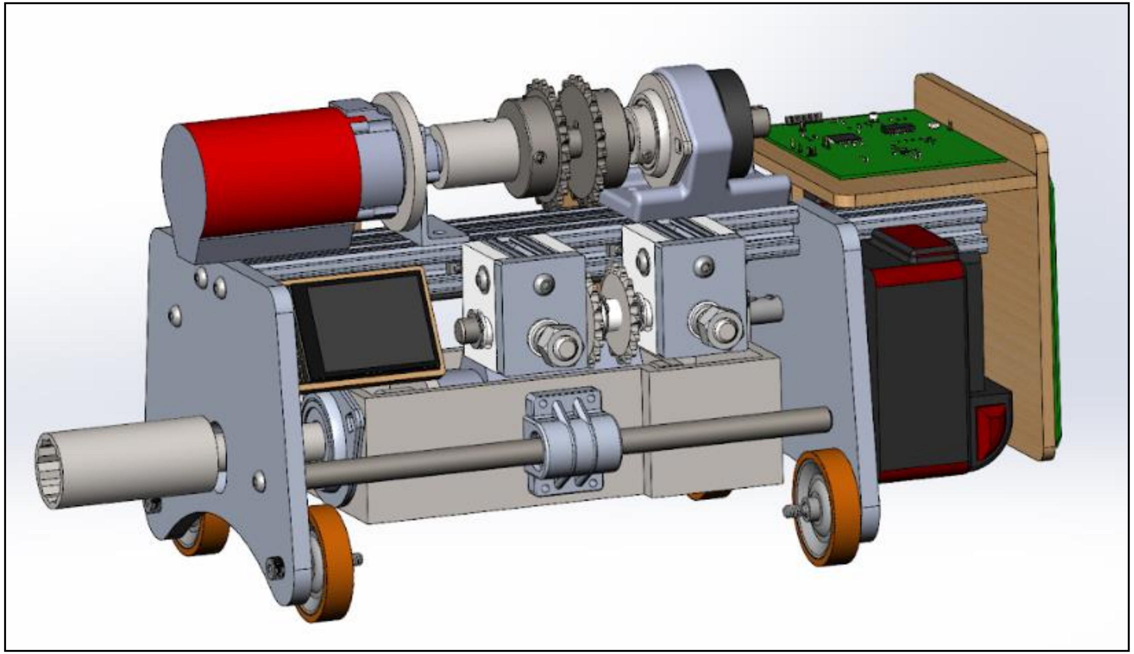
V2



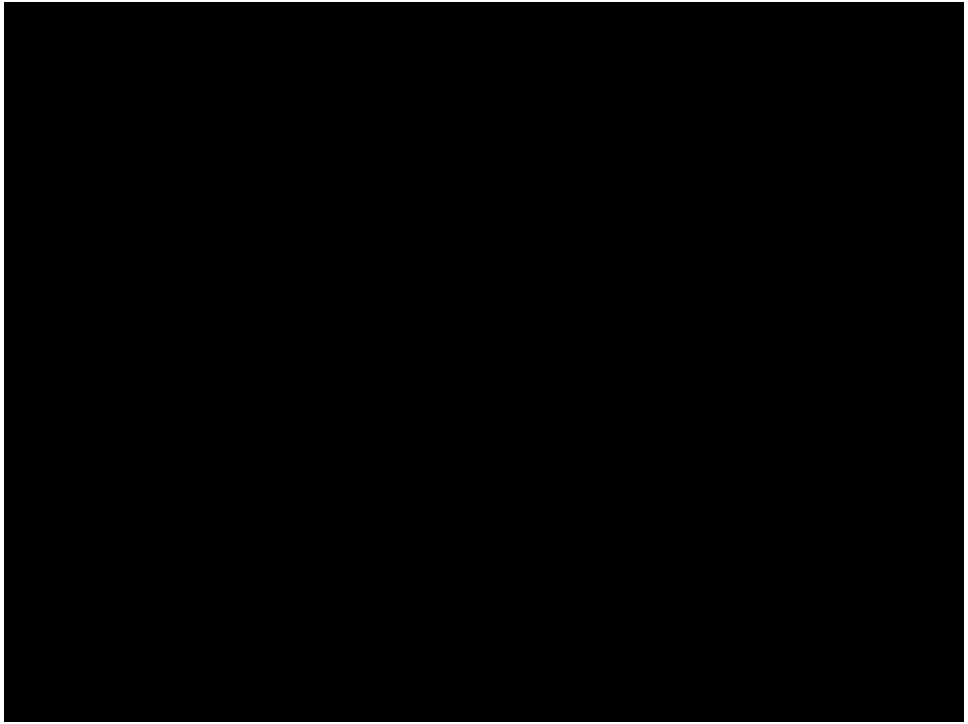
Cobot Evolution - V4

What we set out to do:

- Reach torque up to 40 ft-lbs.
- Rotate around surface of pipe
- Able to align to bolt
- Interface socket with nut
- Software foundation
- Adapt to different pipe sizes

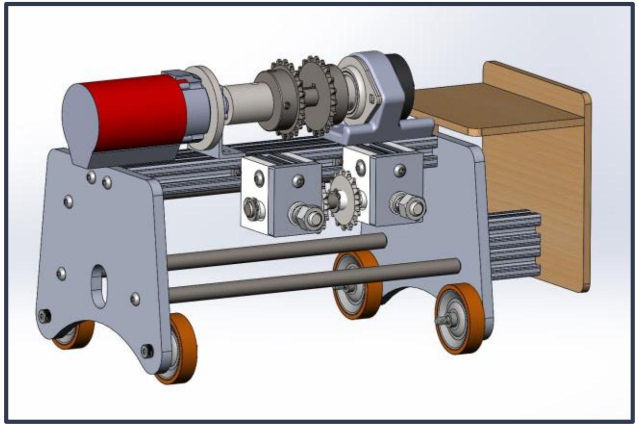


Demo Video of COBOT in Action

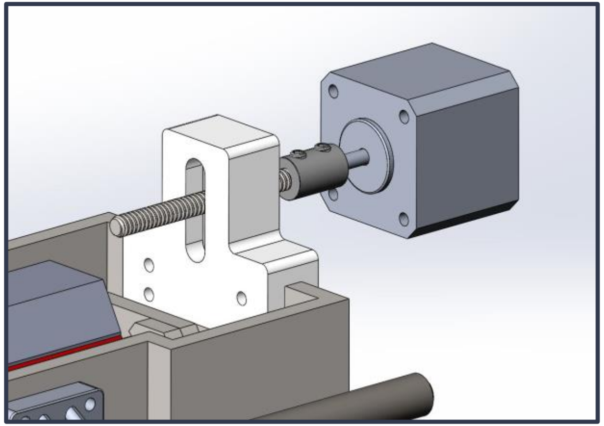


How it works – Mechanical

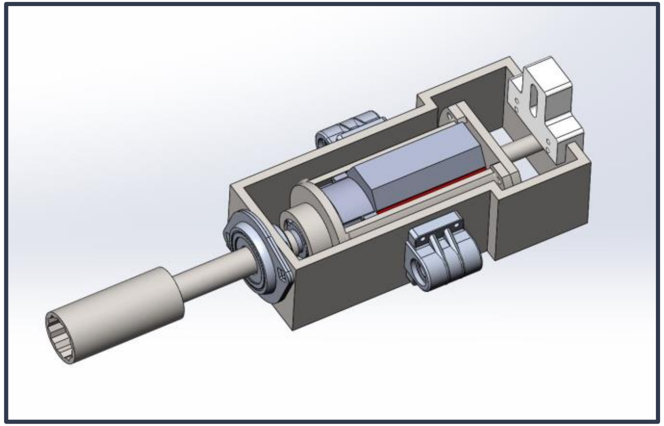
Carriage Subsystem



Articulation Subsystem

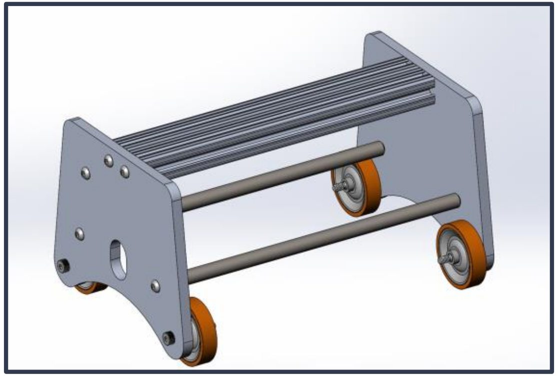


Torque Drive Subsystem



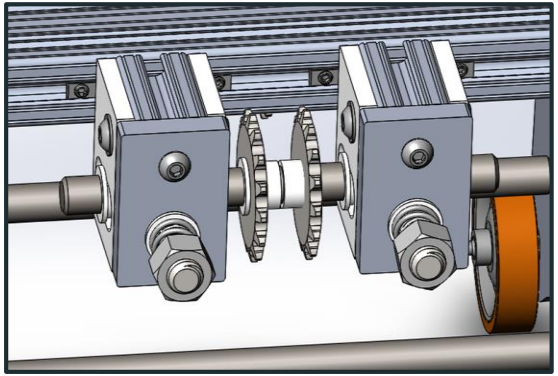
Carriage Subsystem

Frame



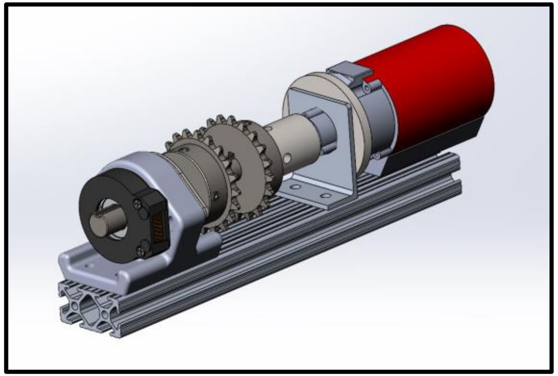
Support and mounting for other systems

Chain Tension Mechanism



Facilitate attachment to pipe

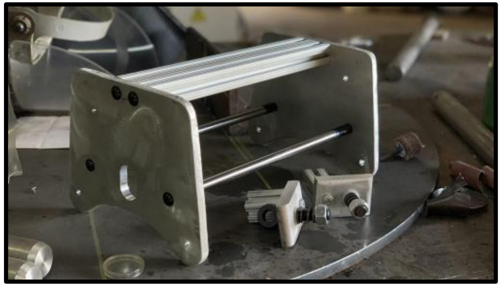
Carriage Main Drive

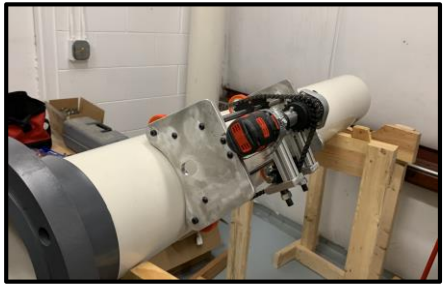


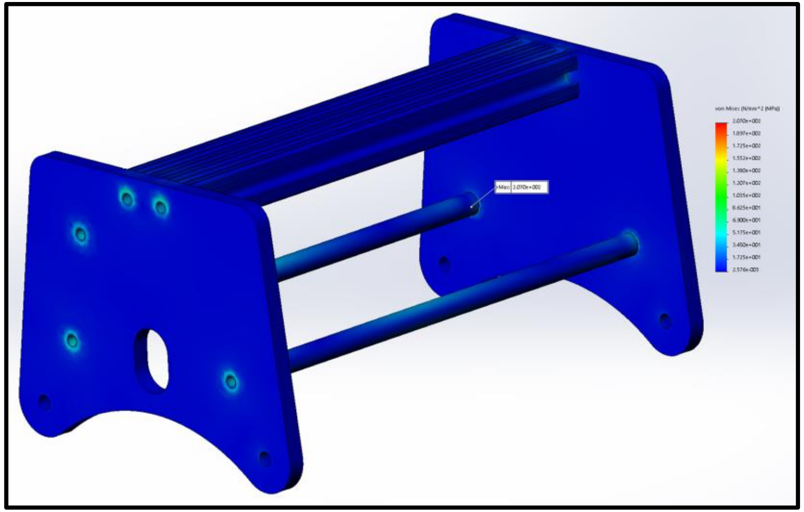
Circumnavigate around the pipe

Carriage Subsystem – Frame

- Main support for components
- Designed for adaptability and future upgrades
  - Larger pipe sizes
  - Increased torque
  - Safety measures
- Able to withstand the 40 ft.lbs torque reaction



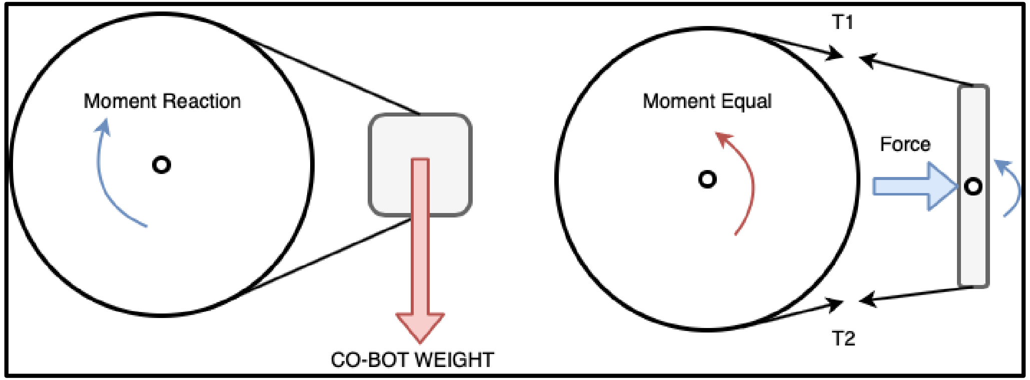




Simulation analysis for torque reaction stress and deformation, shear in bolts, horizontal support beam stress

Carriage Subsystem – Tension

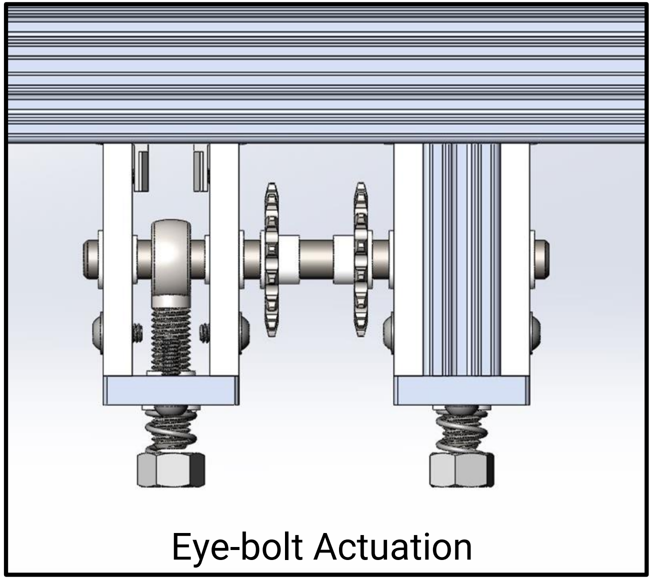
- Easily allow operator to attach the device to pipe
- Provide necessary tension to prevent slippage
    - Tension in chain causes friction

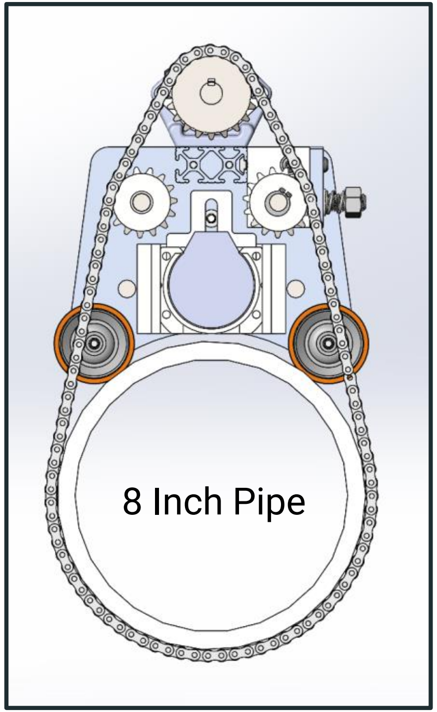


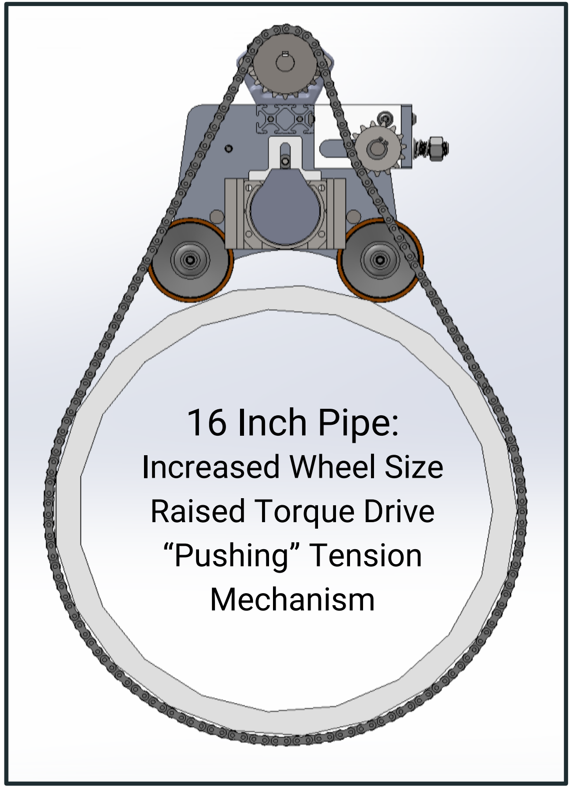
Problem Initially modeled as a drum-brake, simple prototype used

Carriage Subsystem – Tension

- Designed for pipe size adaptability and upgrades

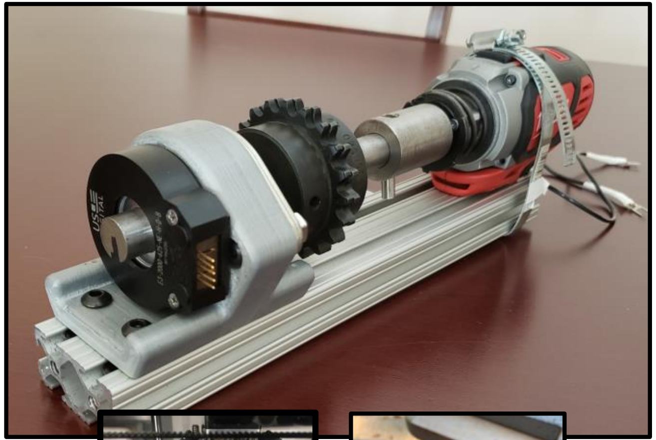


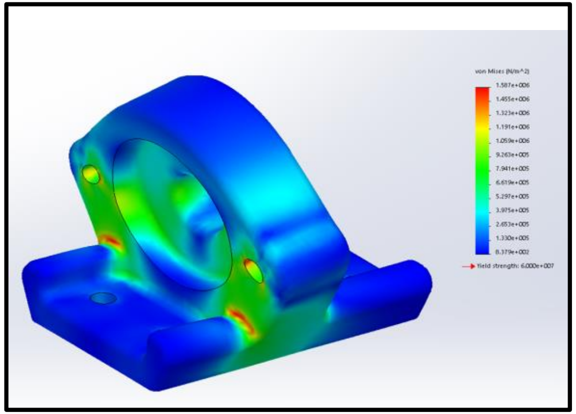


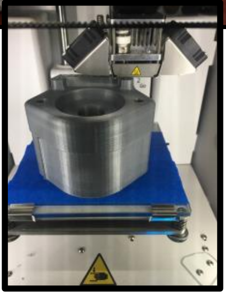


Carriage Subsystem – Main Drive

- Circumnavigating the pipe and indexing to bolts
  - Drive motor, custom coupler, dual ANSI sprockets, custom bearing mount, encoder
- Bearing mount 3D printed
  - Solidworks Analysis



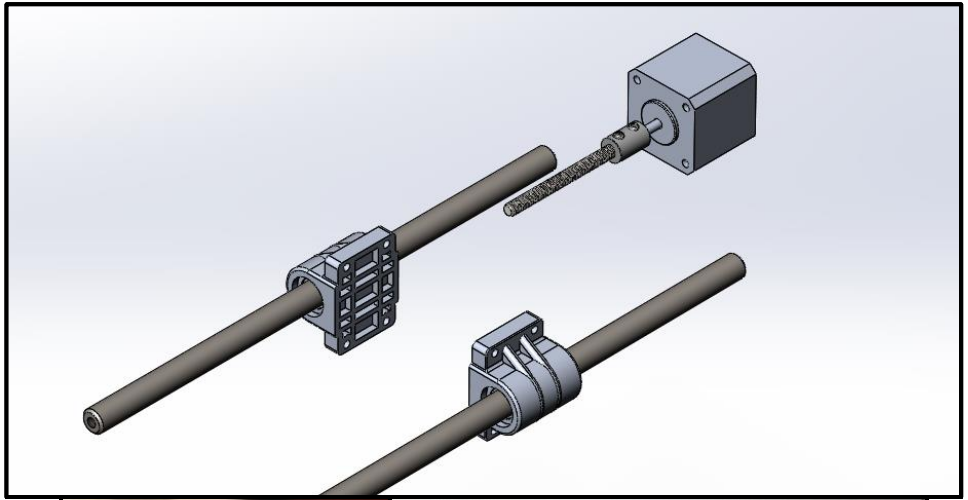


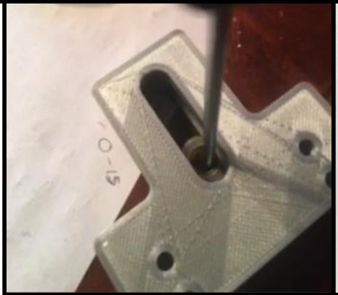


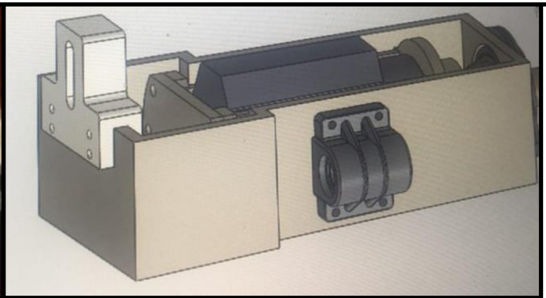
Calculations for holding torque, climbing torque, required shaft thickness, ideal sprocket size, encoder values

Articulation Subsystem

- Integrates subsystems
    - Provides precise controlled linear motion to Torque Drive
    - Supports and transfers loads to the steel shafts of Carriage
    - Adaptable

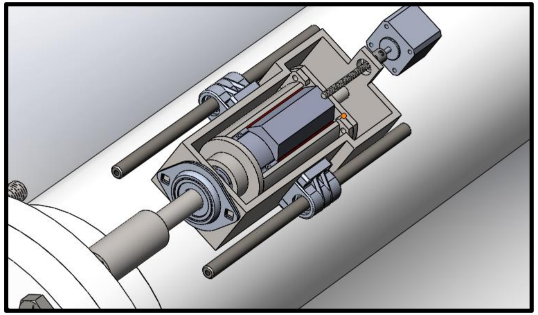




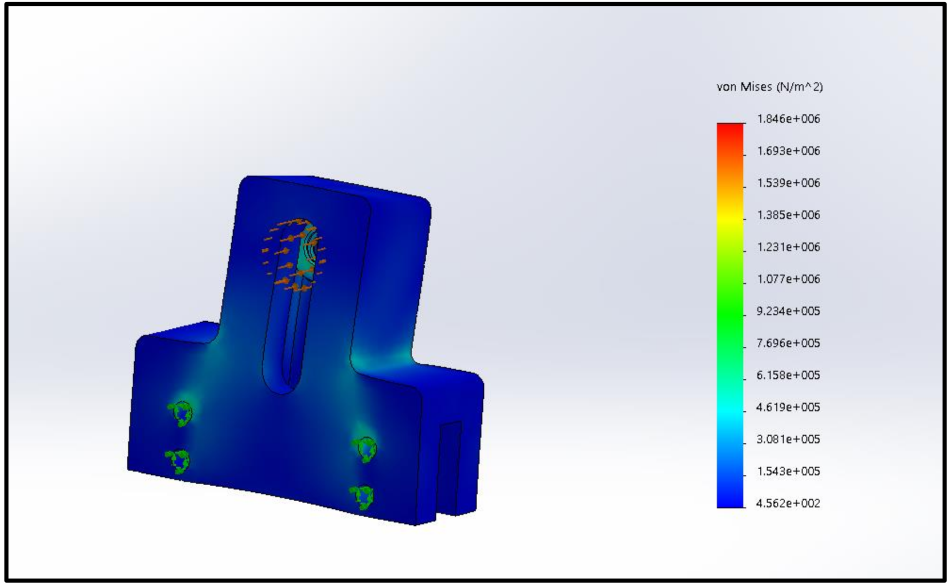


Articulation Calculations and Analysis

- Torque analysis for stepper motor
  - Mechanics of lead screw*: $T_R = \frac{F d_m}{2}\left(\frac{l + \pi f d_m}{\pi d_m - f l}\right)$
  - Tr = 0.96 lb*in or 10.8 N*cm
  - Stepper Motor: Nema 17 (45 N*cm)



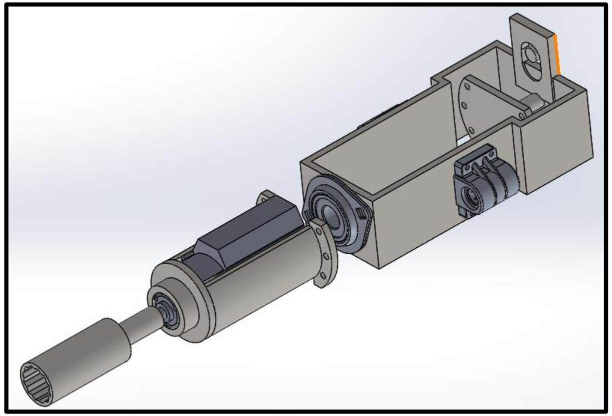
- 3D Printed torque drive and nut connector
  - Tensile Strength of PLA printed at full infill: 30 MPa
  - Static analysis found stresses of 1.5 MPa or below

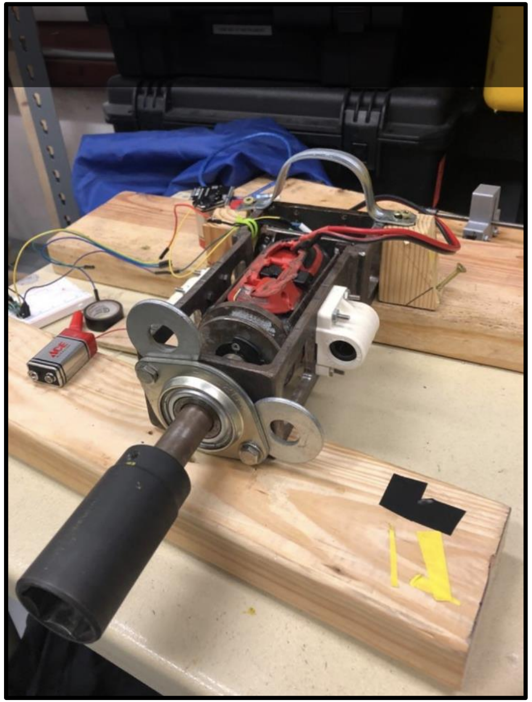


*Source: Shigley's Mechanical Engineering Design, 9th Ed, J. Keith Nisbett and Richard G. Budynas

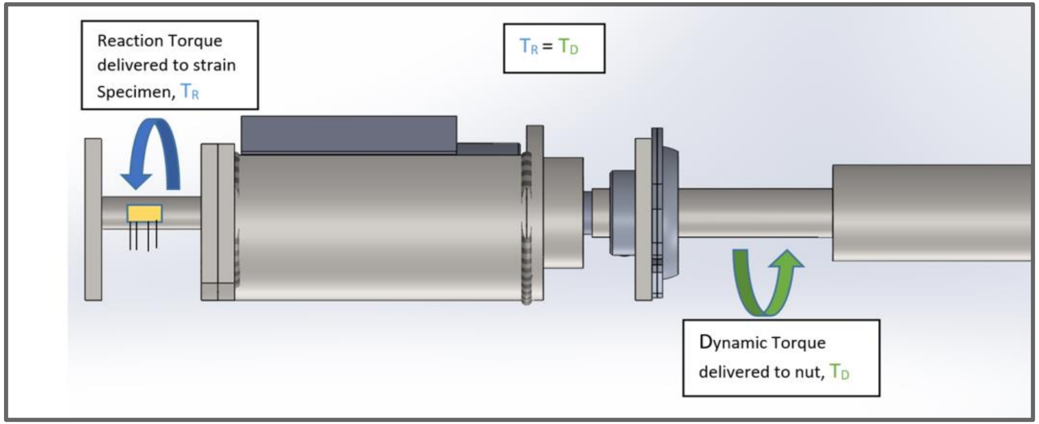
Torque Drive Subsystem

- Tighten and Loosen bolts
  - Motor controller
  - DC Motor
  - Outer Support Housing
  - Socket Extension
- Verify Output Torque
  - Full bridge Strain Gage
  - Free Floating Inner Torque Motor Housing





Torque Drive Subsystem- Torque Measuring Concept



Working Principle

- The dynamic torque delivered to the nut is equal and opposite to that of the reaction torque
- Easier to measure the reaction torque and can be done with a simple strain gage
- ΔTorque → ΔStrain → ΔVoltage → Microcontroller Logic

Test Procedure

- Apply known torque using torque wrench, repeat over a range of values
- Plot corresponding voltage output over this range of torque values
- Repeat and form a calibration curve (see results pg.49)

Torque Drive Subsystem– Designing Test Specimen

Design against plastic deformation

σ' = Von Mises Stress ; S_y = Yield Strength; T=Torque ; J=Polar Moment of Inertia ; τ = Shear Stress ; d = diameter

(1) $\sigma' < S_y$; Stress must be less than yield Strength

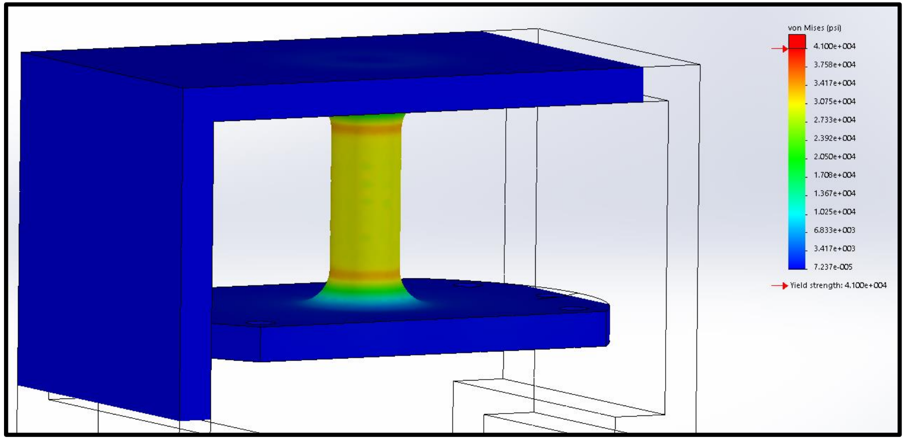
(2) Von Mises: $\sigma' = (\sigma_x^2 - \sigma_x\sigma_y + \sigma_y^2 + 3\tau_{xy}^2)^{1/2}$

(3) $\tau = (T*d/J*2)$ and $\sigma_x = \sigma_y = 0$

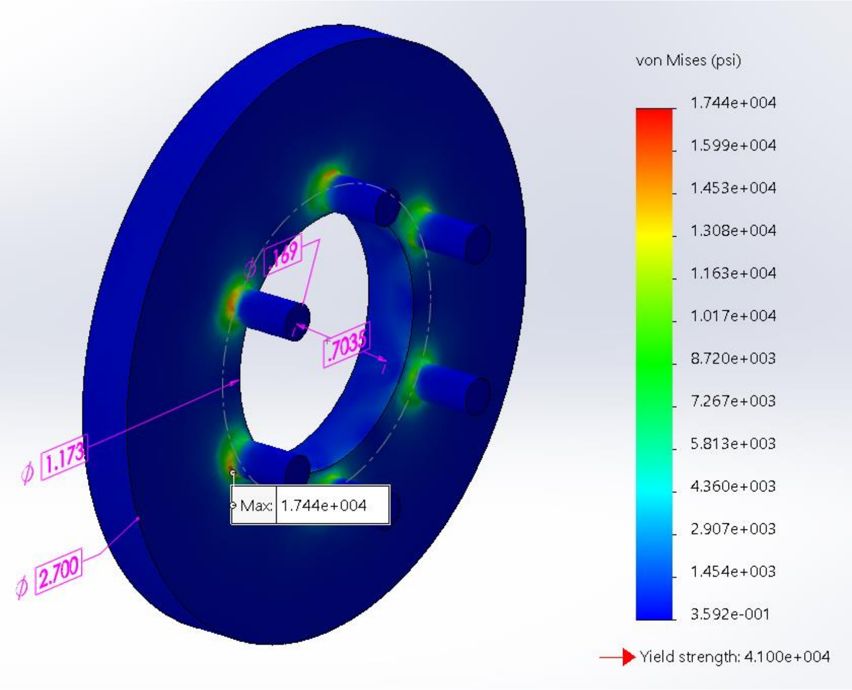
T (ft*lb)	D (in)	S_y (ksi)	σ'	F.O.S
40	0.5	40.1 (low carbon steel)	33.8	1.2

Goals

- Repeatable Results
  - No plastic deformation
- Geometric
  - Strain gage requires radius of curvature > 3mm.
- Easy to work with material
  - Low carbon steel



Torque Drive Subsystem-Drill Clutch Adapter



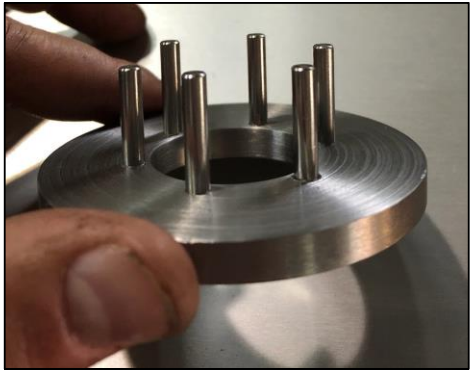
<u>Purpose</u>

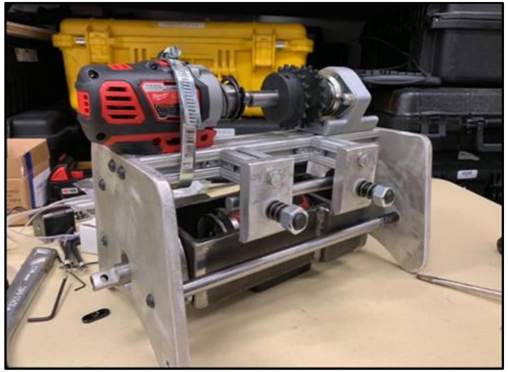
- Rigidly attach drill to inner housing
- Essentially a motor mount
- Keeps drill clutch fully engaged

Manufacturing the Cobot









List of Components – Electrical/Computer

Component Name	**Specs & Features**	**Purpose**
Arduino Mega 2560	Additional pins are available	Support a flexible design approach
Cytron Motor Controller (2)	Current rating compatible with the high-current draw of the drill motors	Drive the M18 Drill Motors in Carriage and Torque Drive Subsystems
Easy Driver Stepper Motor Controller	Compatibility and user out-of-the-box implementation	Drive the Nema 17 Stepper Motor in the Articulation Subsystem

Software Functions for Subsystems

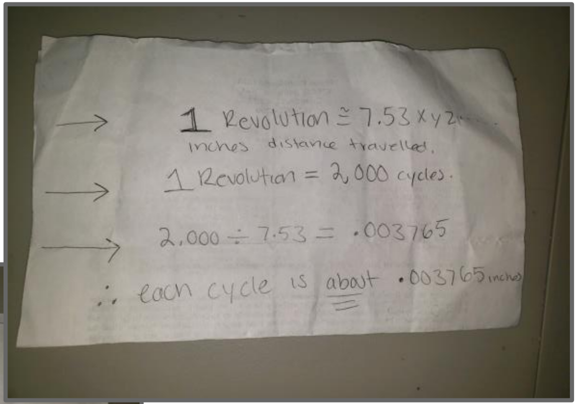
- activateTorqueMotor()
  - First Subassembly to run
  - Begins the process of tightening the bolt
- articulationSystemBackward()
  - Second Subassembly to run
  - Disengages the Torque Drive subsystem from the bolt
- startCarriageMotor()
  - Third Subassembly to run
  - Begins the process of circumnavigating around The pipe to find next bolt
- articulationSystemForward()
  - Fourth Subassembly to run
  - Engages the Torque Drive subsystem towards the bolt

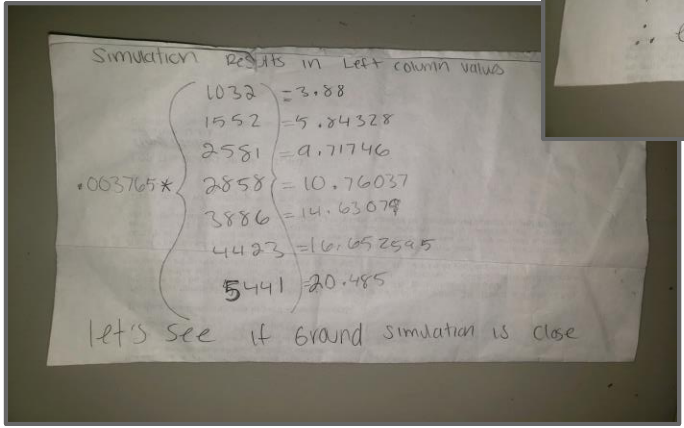
```
void articulationSystemBackward()
{
  Serial.println("Articulation System Activated");

  myStepper.setSpeed(SRPM);
  for(int i = 0; i < REVOLUTIONS; i++)
  {
          myStepper.step(-stepsPerRevolution);
          delayMicroseconds(DELAYMCRO);

  }
startCarriageMotor();

}

void startCarriageMotor()
 {
```

PLanning Out Software Logic

- Identify coordinates of bolts
- Using circumference of pipe to find distance
- Using peak diameter of gear and CPR for estimated travel
- Plan out path for torque sequencing
- Use Computer Science teammate suggestion for one direction travel

Multiply Cycles by estimated constant to define a distance





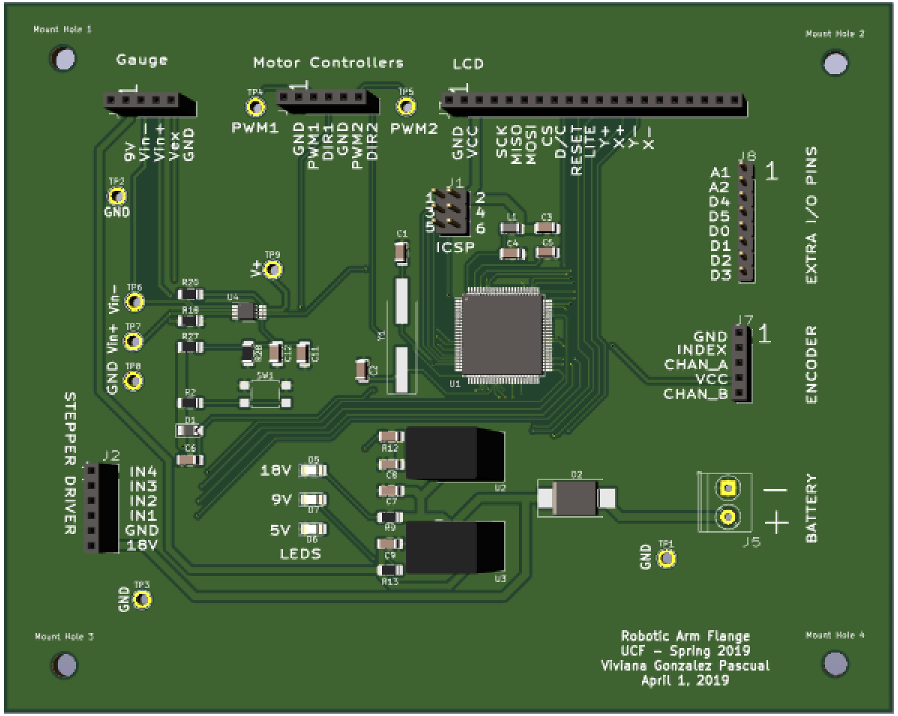
How it works - Control Logic

- The success of the COBOT carrying out all of the logic depends on the success of the Carriage Subsystem control logic
    - Carriage is responsible for driving to each bolt
    - Code snippet demonstrates how data received from the encoder is used as the signal for PID logic in software
    - PID is a proportional–integral–derivative controller automatically applies accurate and responsive correction to a control function

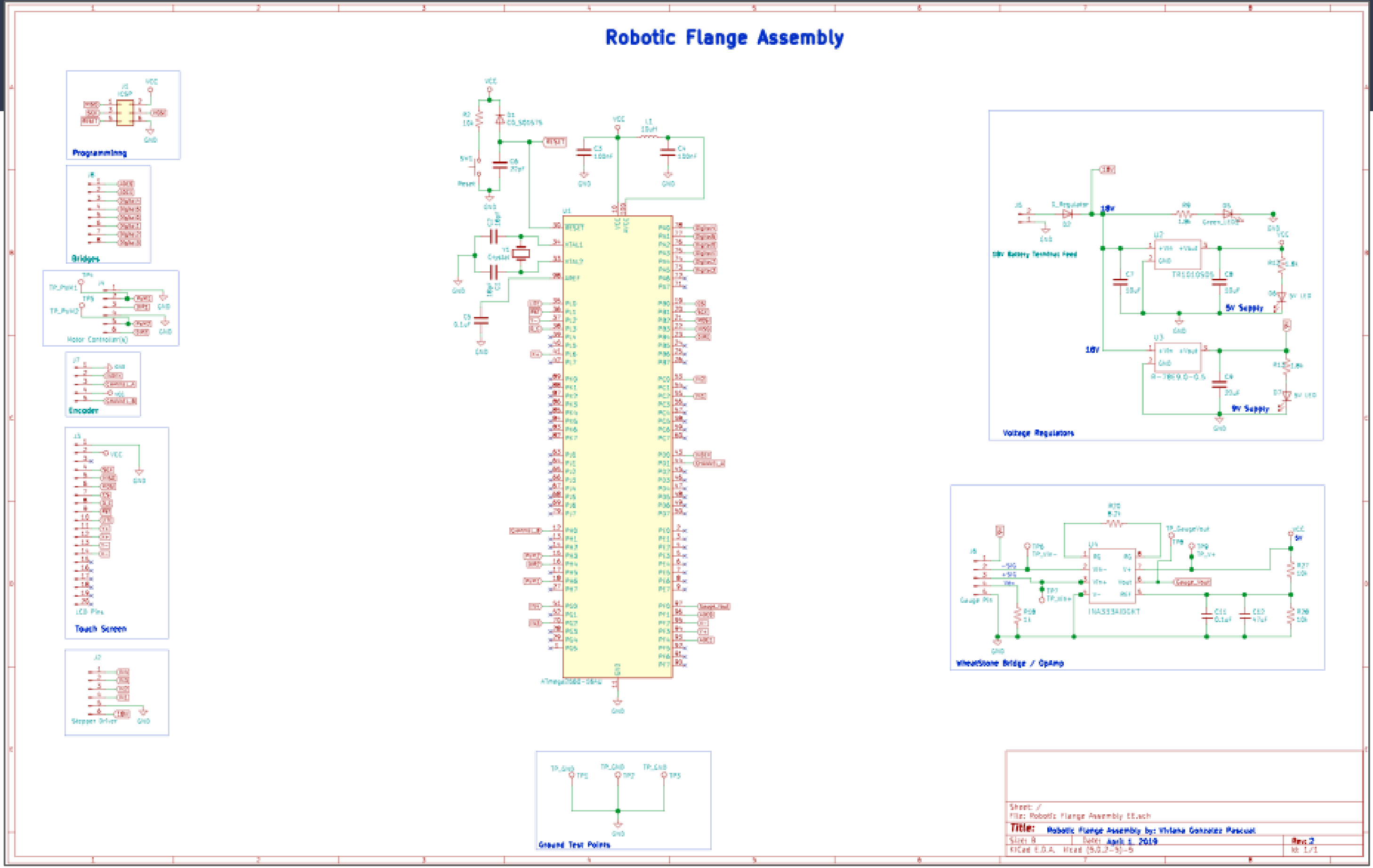
```
if (locationFound)
  {
    // did we really find the location? If not let PID handle it, otherwise we got lucky so here we are
    if(abs(map(encoder0Position, 0, 2000, 0, MAPval)) == positions[next])
    {
      // reset location found and disable other functions that are not desired when we are at a bolt location
      locationFound = false;
      keepLooking = false;

      // dont go out of bounds in our positions array! C be dangerous
      next = next == 7 ? 0 : (next + 1);

      if(!stopForever)
      {

      activateTorqueMotor();
      }
    }

  // This updates PID signal so that PID can PID in a more PID sorta way
  // (input gets mathed and then output tells PID stuff)
  input = encoder0Position;

  myPID.Compute();  // calculate new output

  // If stopForever is true the COBOT must be done
  if(stopForever)
  {
    carriageMotor.setSpeed(0);
  }
// only do this stuff if COBOT is not busy with other stuff
if(keepLooking == true)
{
  setpoint = positions[next];
    pwmOut(output);
}
```

How it works – Electrical & Computer

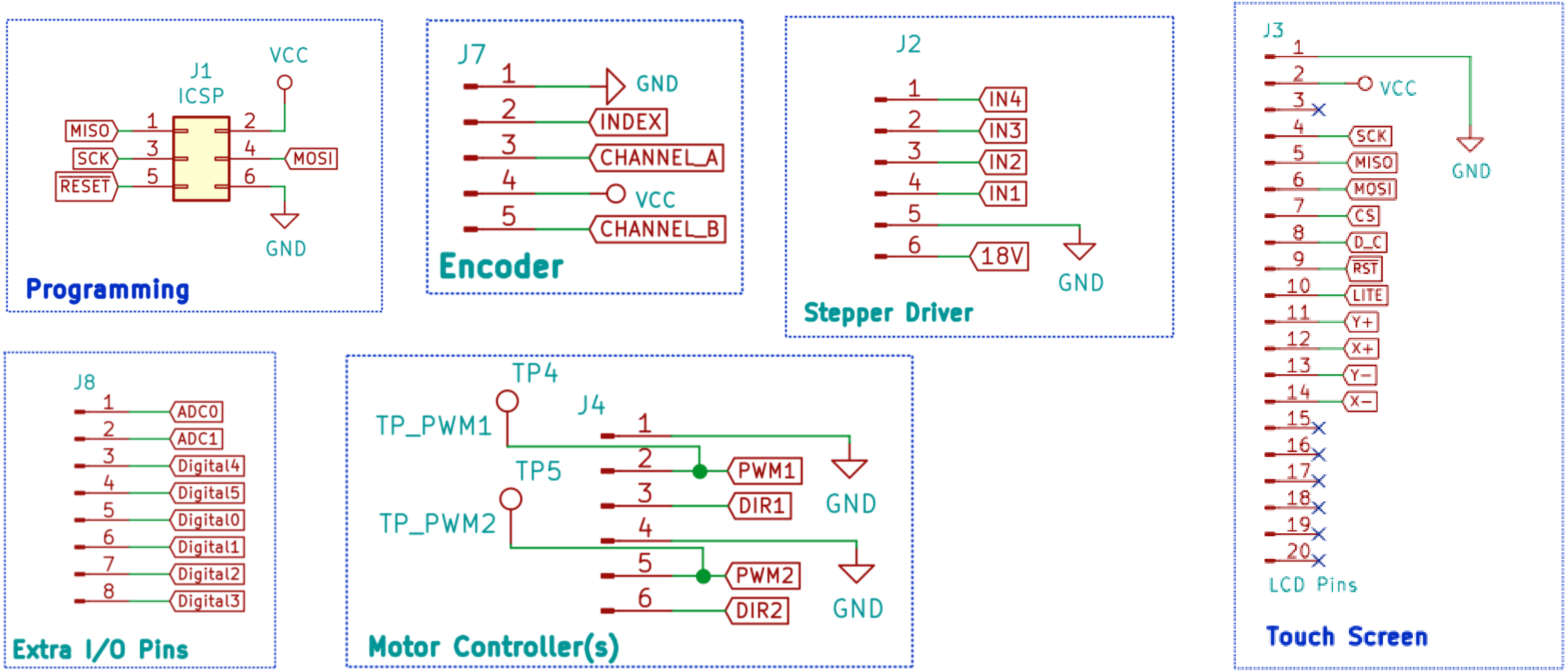
- The interface between the mechanical hardware and electrical components.
  - Provides COBOT with logical control
  - Implements main electrical connections to subsystems
  - Regulates power given to certain components



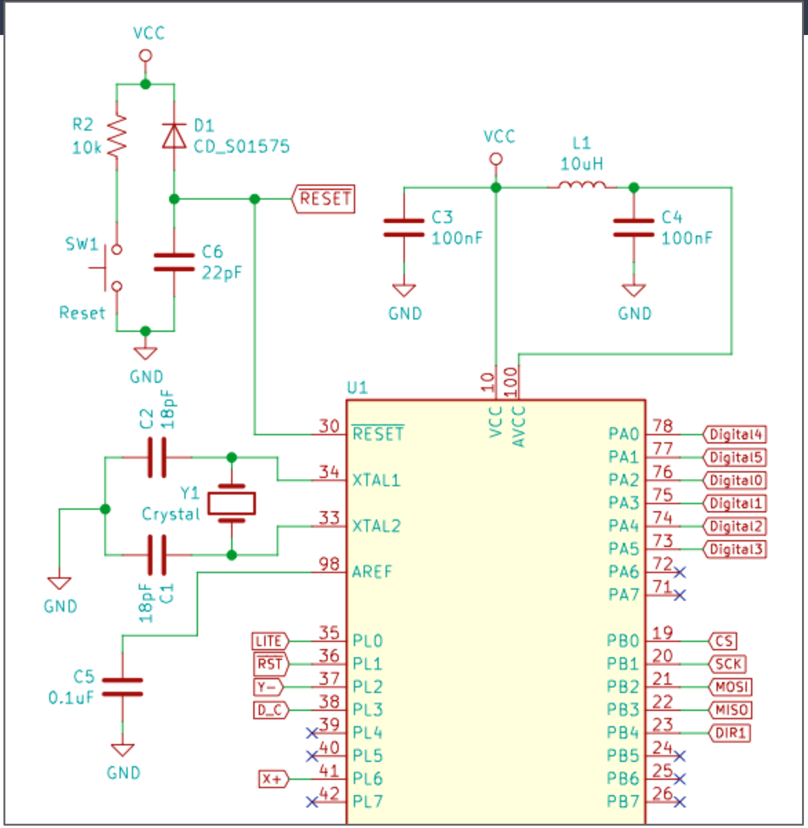
Printed Circuit Board (PCB) Schematic

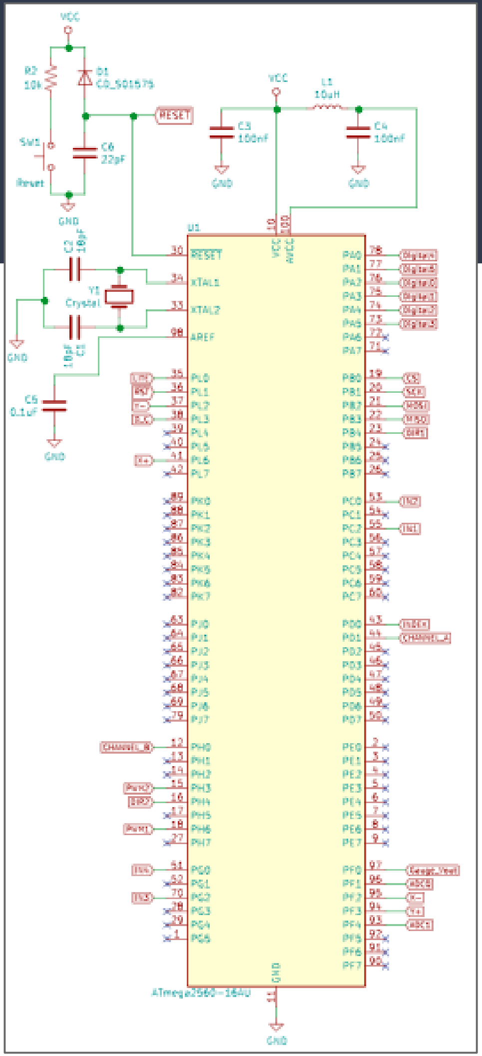


Input/Output Pins

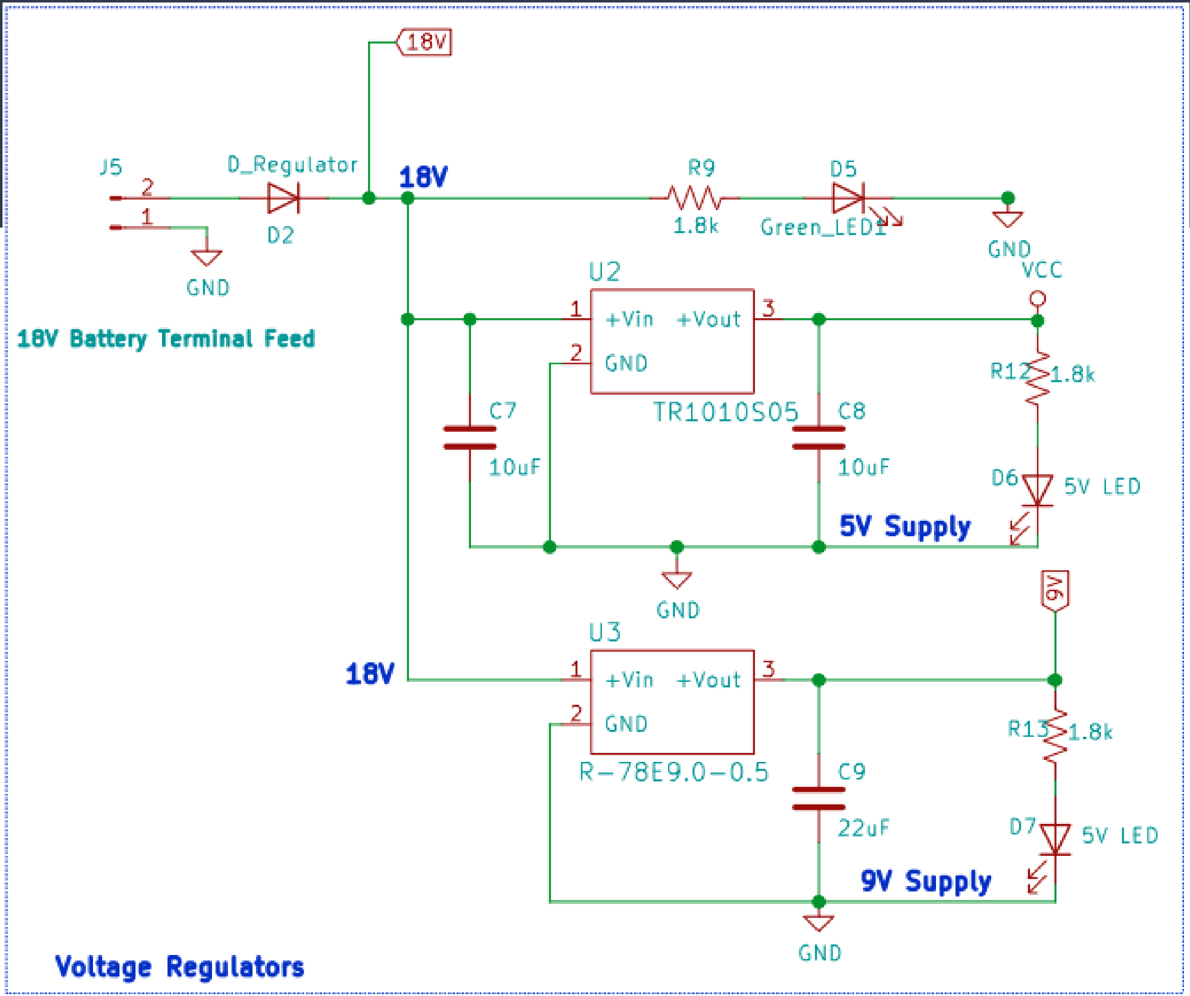


Multipoint Control Unit (MCU)

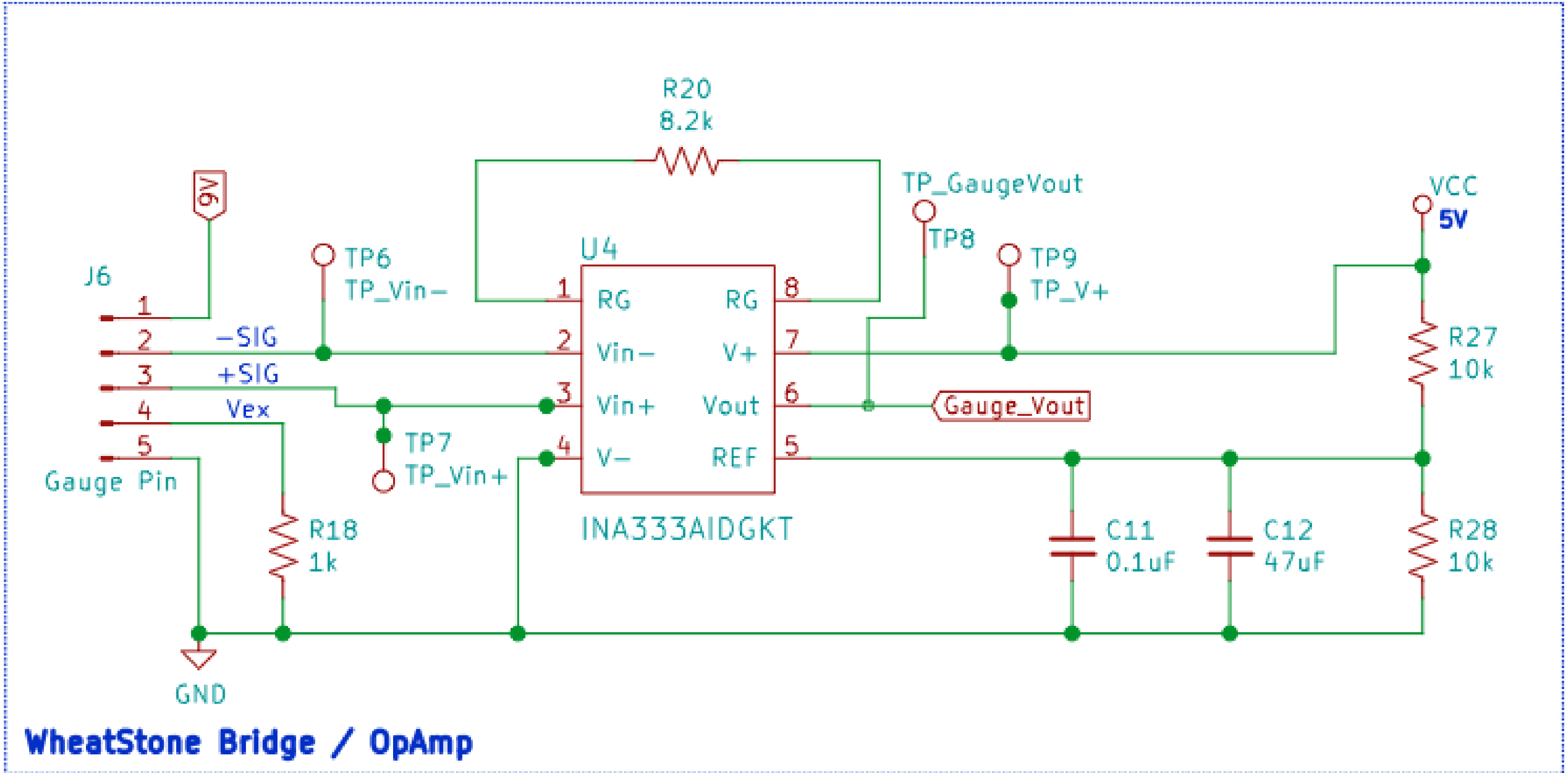




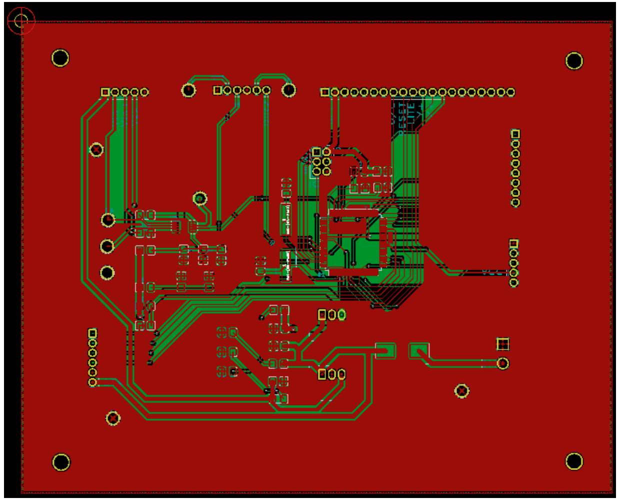
Voltage Regulators

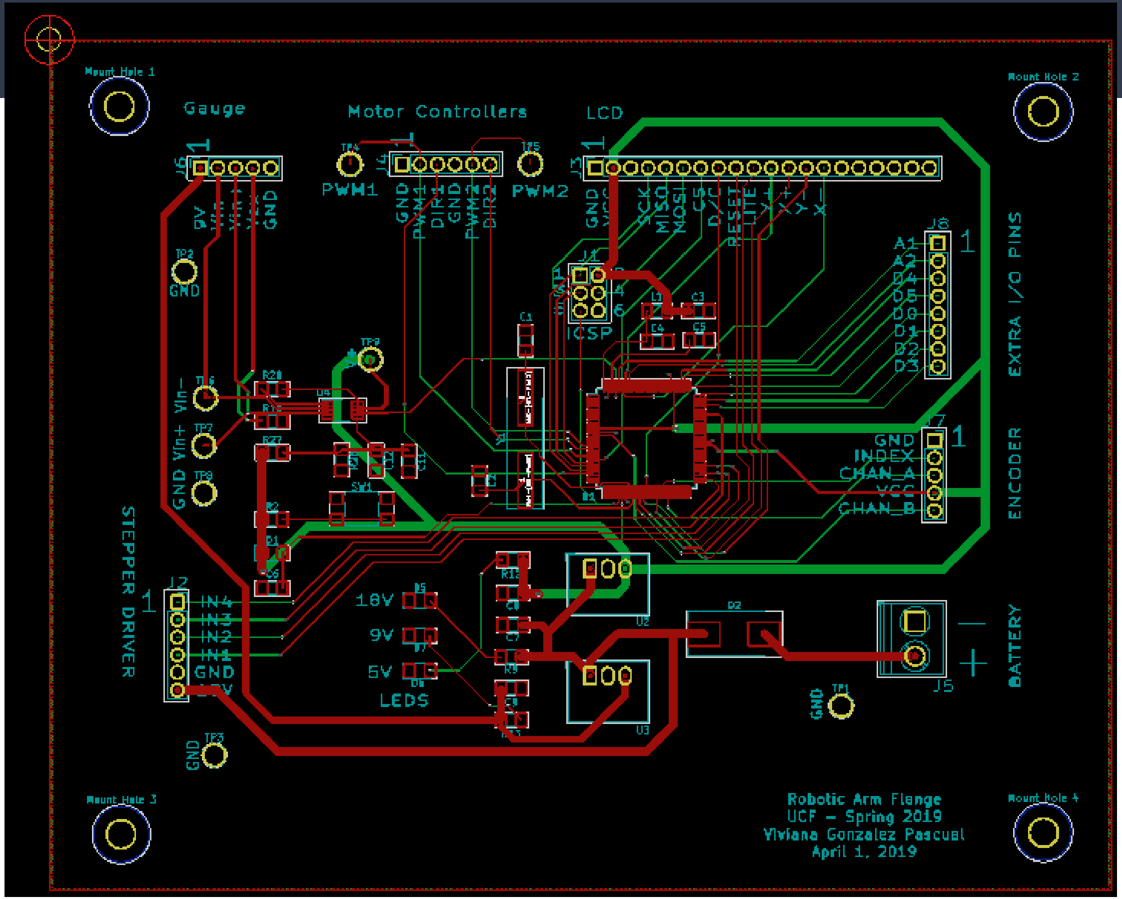


Strain Gauge



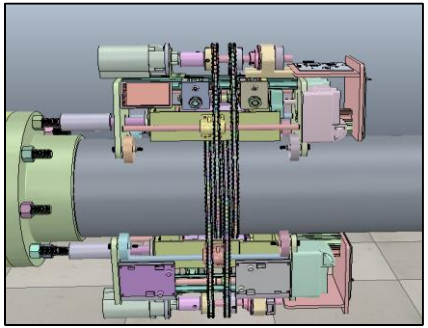
Printed Circuit Board (PCB) Final Layout



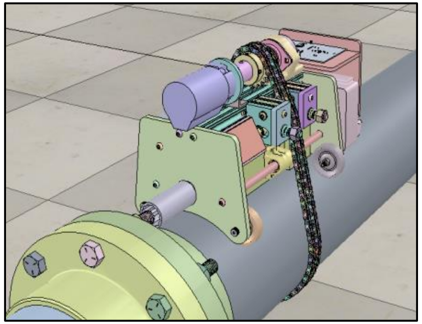


Simulation Scenarios

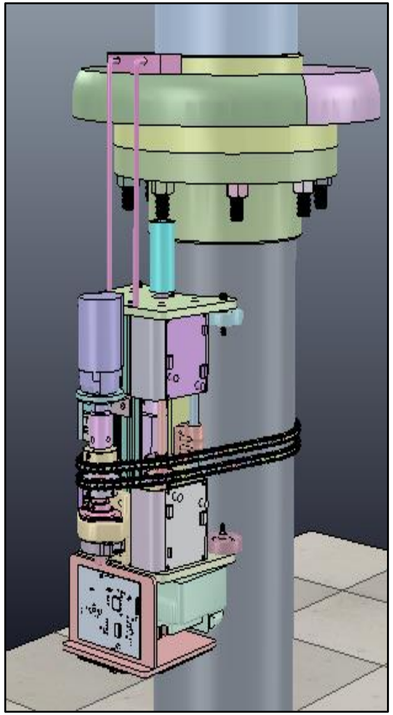
2-Cobot



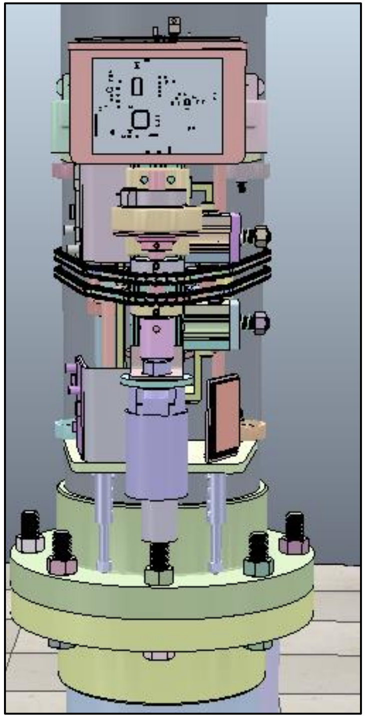
Proto-Cobot



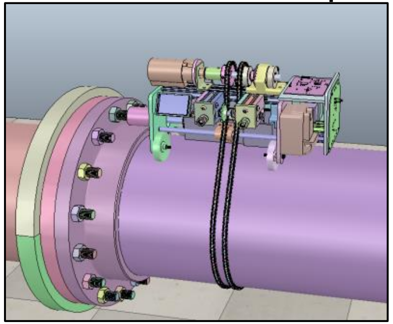
Upward Design



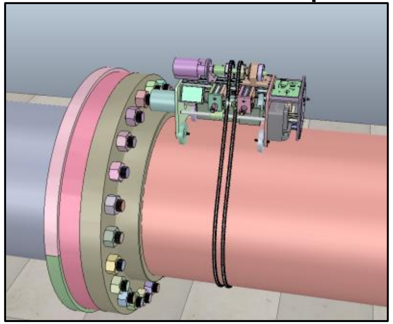
Downward Design



16-bolt Scale-up



24-bolt Scale-up



Features

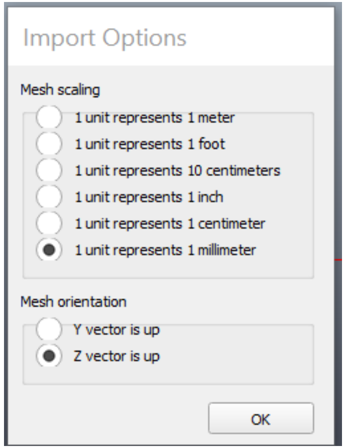
- Finished
  - Full dynamic movement
  - Socket alignment
  - Torque drive positioning
  - Proximity sensors
  - Camera view
  - Data log
  - Expandable database
  - User Interface
    - User options
    - Dynamic results
    - Live progress
    - Estimated time to completion
- Future
  - Torque calibration
  - Precise time estimate
  - Alignment via camera
  - Perform in depth safety analysis
  - Test limitations of design

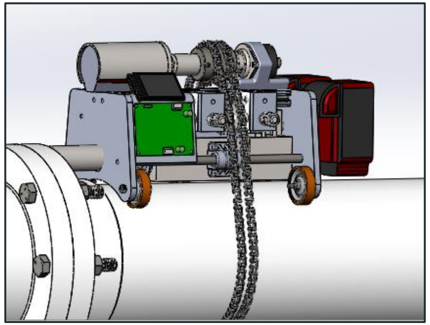
V-REP & The Build

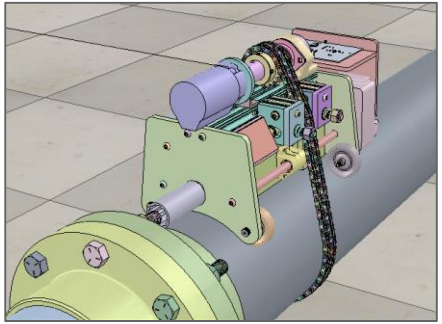
- V-REP (Virtual Robot Experimentation Platform): A highly customizable robotic simulator.

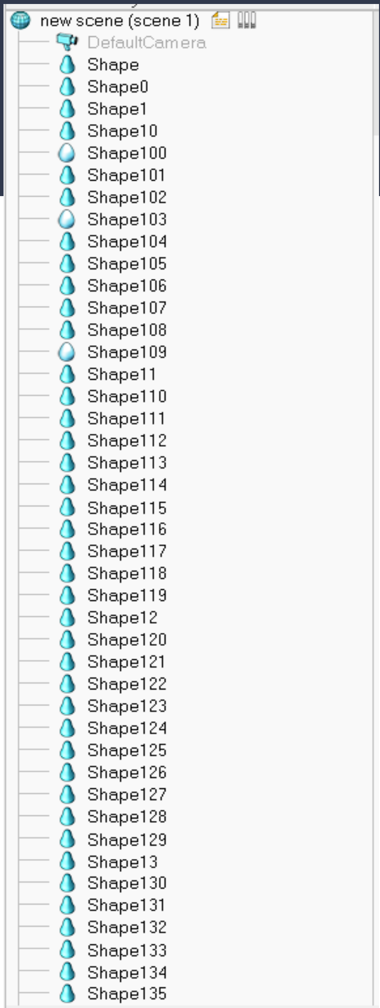
Build:

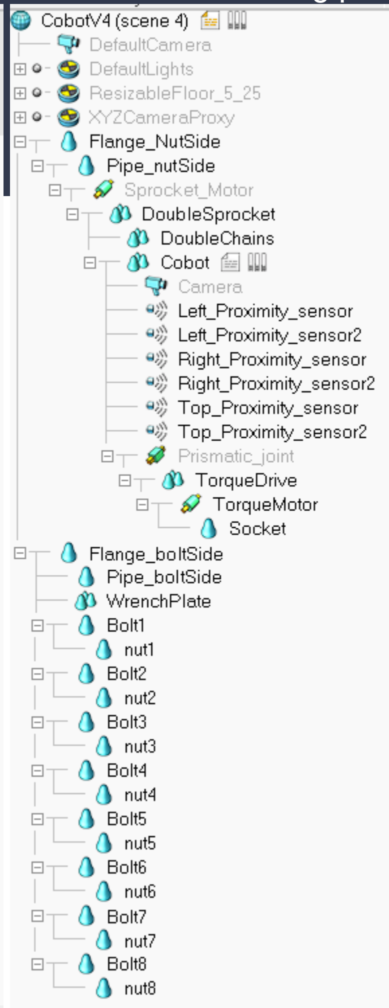
- Save files in SolidWorks as STL.
- Imported all STL files into V-REP as mesh.
- Select the desired scaling and orientation.
- Each component would then have to be labeled and grouped accordingly.





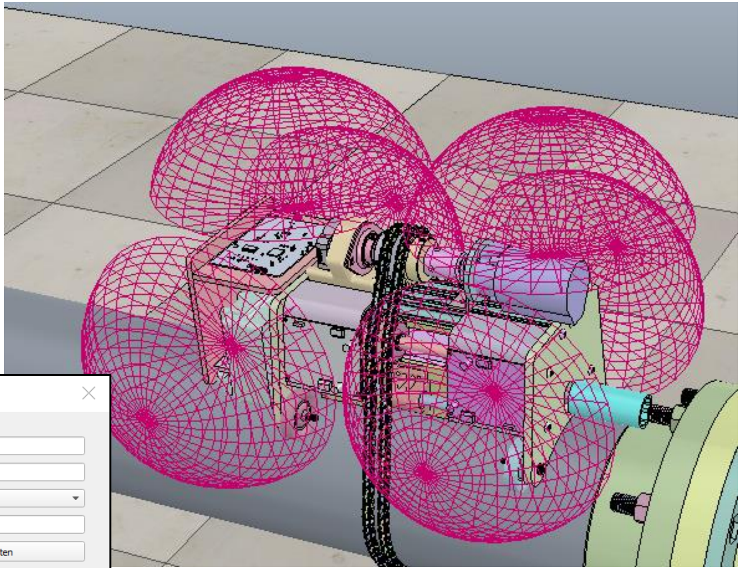


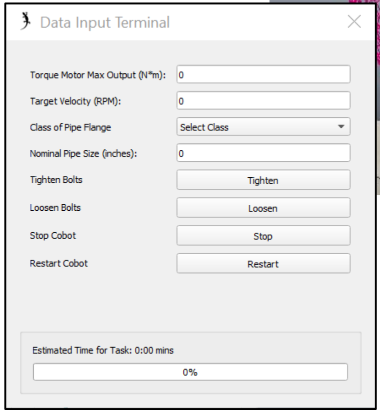




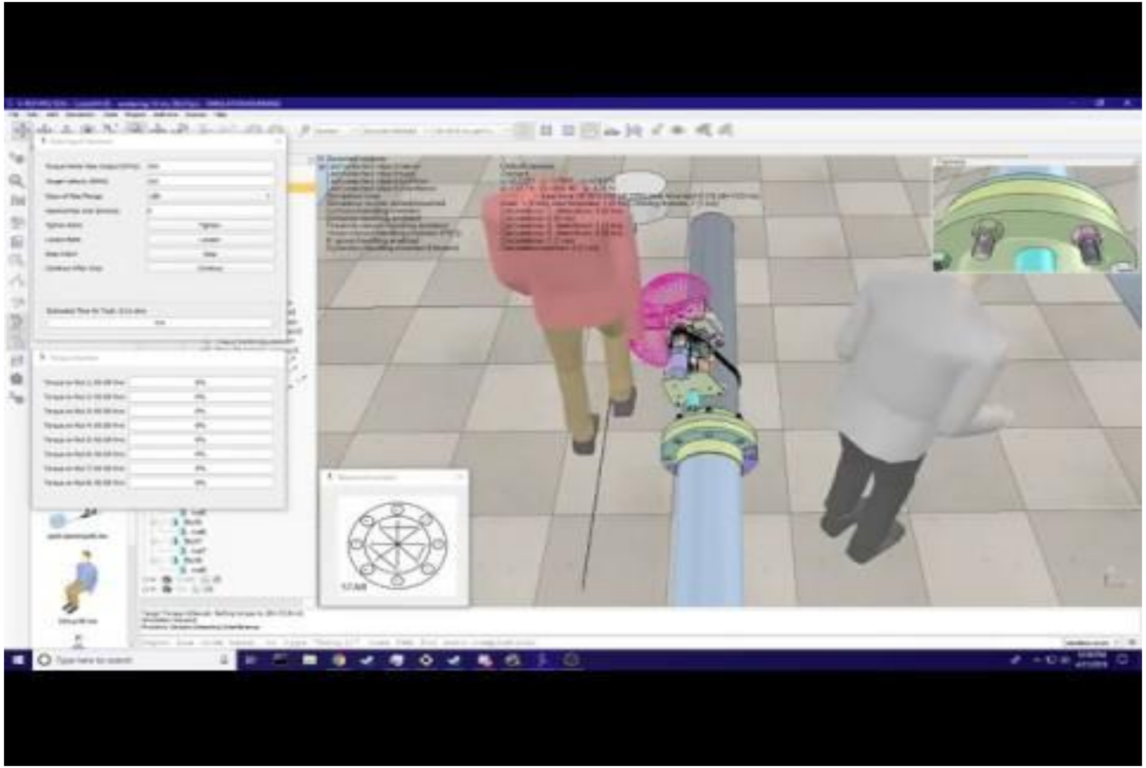
Safety

- Proximity sensors installed on the left, right, and top of the Cobot
  - Sensors check for objects in the nearby proximity of the Cobot and stop the Cobot when a collision is detected
- Emergency shutdown features enabled for when the proximity sensors detect anomalies
- Stop button in UI as backup
- Cobot programmed to return to starting position after completion of its task and shuts off



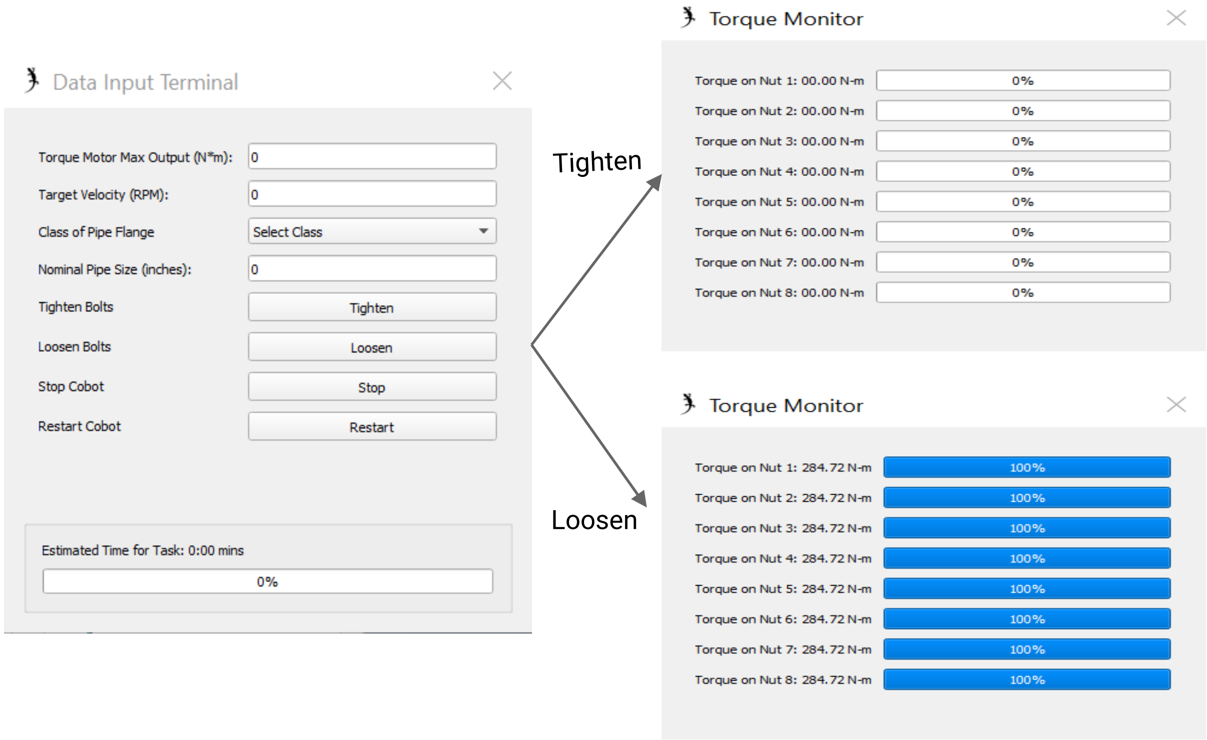


Safety Demo



User Interface

- 4 Inputs
  - Max Torque
  - Max RPM
  - Class of Pipe
  - Nominal Size
- 4 Buttons
  - Tighten
  - Loosen
  - Stop
  - Restart
- Pop-up Window
  - Depends on Button
  - Progress Bars



Estimated Time

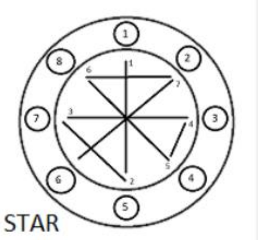
$$n = \left(\frac{numberBolts}{8}\right) + 1$$

$$TimeToRevolve = \sum_{i=0}^{n} 2^{(n-i)} * \frac{2}{2^{i}}$$

180° 90°

45°

4 2 1

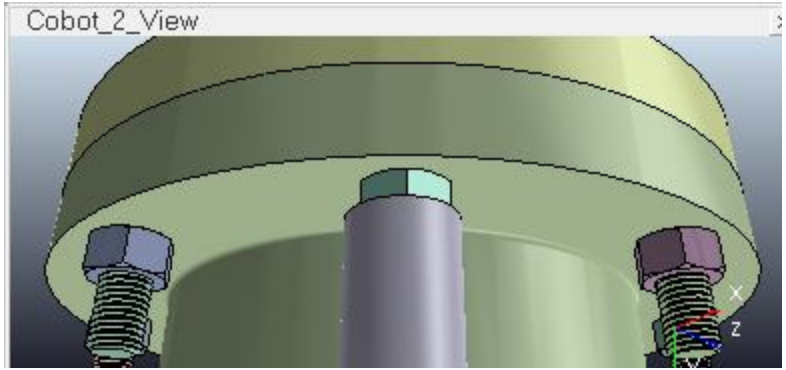


Estimated Time = (TimeToRevolve + TimeForDelay + TimeTorqueing) * numberIterations

```
for l = loop, 0, -1 do
    timeToRevolve = timeToRevolve + ((2 ^ l) * (2 / (2 ^ count)))
    count = count + 1
end

timeForDelay = numberBolts * 2

local estimatedTime = timeForBolts + ((timeToRevolve + timeForDelay) * 3)
```

Alignment

- Threshold: A small error used to compare the smallest difference in the coordinates of the cobot socket and the next intended bolt.
- Change in Speed: As the cobot simulation comes near the bolt, it gradually slows down until the second threshold value for the two coordinates are reached




```
tmp = 0.03
tmp2 = 0.001
    if i <= numberBolts/2 then
        if (b[1]-coordinates[i][1] < tmp and b[1]-coordinates[i][1] > -tmp) and (b[3]-coordinates[i][3] < tmp and b[3]-coordinates[i][3] > -tmp) then
        -- Torque the nut
        if wait == false then
             sim.setJointTargetVelocity(revMotor,0.15)
        end

        if (b[1]-coordinates[i][1] < tmp2 and b[1]-coordinates[i][1] > -tmp2) and (b[3]-coordinates[i][3] < tmp2 and b[3]-coordinates[i][3] > -tmp2) then
            sim.setJointTargetVelocity(revMotor,0)
            wait = true
            boltAlign = false
        end
```

Database & Log

- Internal database holds number of bolts and require torque for each nominal pipe size in the different classes
- Pipe size and flange class are taken as inputs from the UI and the database returns the number of bolts and required torque
- Log: Contains the selected inputs from UI. Torque on each nut at a given time along with the applied torque for 3 passes (33%, 66%, 100%). Possible errors also logged.

```
---------------------------------------------
Starting Simulation: Mon Apr  8 13:52:26 2019
Class Selected:
150
Number of bolts:
8
Require Torque (N-m):
284.72
Target Torque Achieved. Setting torque to 284.72 (N-m)
12.95 s: Data inputed into UI

Starting Tightening Sequence

33% Torque:
0.35 s: Torqueing Nut 1
Torqued to: 96 N-m
7.15 s: Torqueing Nut 5
Torqued to: 96 N-m
13.75 s: Torqueing Nut 7
Torqued to: 96 N-m
20.45 s: Torqueing Nut 3
Torqued to: 96 N-m
26.65 s: Torqueing Nut 4
```

```
--assign values for class 150 pipe
function checkClass150(value)
    --Class 150 Dimensions
    C150sizep25To1p5 = {numBolts = 4, requireTorque = 81.35}
    C150size2To3 = {numBolts = 4, requireTorque = 162.70}
    C150size3p5To4 = {numBolts = 8, requireTorque = 162.70}
    C150size5To8 = {numBolts = 8, requireTorque = 284.72}
    C150size10To12 = {numBolts = 12, requireTorque = 474.54}
    C150size14 = {numBolts = 12, requireTorque = 677.91}
    C150size16 = {numBolts = 16, requireTorque = 677.91}
    C150size18 = {numBolts = 16, requireTorque = 1016.86}
    C150size20 = {numBolts = 20, requireTorque = 1016.86}
    C150size24 = {numBolts = 20, requireTorque = 1423.61}

    -- Log matched values to the log and call other functions to compare torque, create torque monitor UI and it initiate movement pattern.
    if checkDatabaseOnce == true then
        if value == "0.25" or value == "0.5" or value == "0.75" or value == "1" or value == "1.25" or value == "1.5" then
            print("Number of bolts:")
            print(C150sizep25To1p5.numBolts)
            writeToFile("Number of bolts:")
            writeToFile(tostring(C150sizep25To1p5.numBolts))
            print("Require Torque (N-m):")
            print(C150sizep25To1p5.requireTorque)
            writeToFile("Require Torque (N-m):")
            writeToFile(tostring(C150sizep25To1p5.requireTorque))

            checkTorqueTarget(C150sizep25To1p5)
            coordinates=BoltCoordinates(4)
            createTable(4)
            checkDatabaseOnce = false

        elseif value == "2" or value == "2.5" or value == "3" then
            print("Number of bolts:")
            print(C150size2To3.numBolts)
            writeToFile("Number of bolts:")
            writeToFile(tostring(C150size2To3.numBolts))
            print("Require Torque (N-m):")
            print(C150size2To3.requireTorque)
```

Error Handling

- Unreachable torque error: Error occurs when torque value inputted in the UI is lesser than required torque value based on class of flange
- Database error: Error occurs when class of flange does being referenced is not a part of the database
- Movement error: Error occurs when there is no movement pattern for a flange with certain number of bolts
- Proximity error: Error occurs when there is an object that obstructs the movement of the cobot triggering the proximity sensors
- Kill error: Emergency kill error that stops the cobot from moving and torquing
- Restart: Continues moving the cobot around the flange after error has been resolved
- Missing Input: If the UI receives no values from the user, the simulation will show error message stating missing input
- Invalid Input: Non-number values are entered. Pop up will notify of error

```
-- Function handles all error that might occur.
function errorHandling(typeError)
    -- Error is thrown when the input torque is less than the requiar found in the database and user selected "Loosen".
    -- A pop-up window appears to notify user and error is logged and the simulation is suspended.
    if typeError == "unreachableTorque" and loose == true then
        warning = [[
            <ui title="Torque Monitor" closeable="true" resizable="false" placement="center" activate="false">
                <label text="Target Torque Unreachable. Bolts cannot be loosened." id="1"/>

                <label text="" style="* {margin-left: 400px;}"/>
            </ui>
            ]]
                errorui=simUI.create(warning)
        writeToFile("Target Torque Unreachable. Bolts cannot be loosened.")
        print("Target Torque Unreachable. Bolts cannot be loosened.")
        dontRun = true

    -- Behaves like the above, but still allows the simulation to run until input torque value is reached.
    elseif typeError == "unreachableTorque" then
        warning = [[
            <ui title="Torque Monitor" closeable="true" resizable="false" placement="center" activate="false">
                <label text="Target Torque Unreachable." id="1"/>

                <label text="" style="* {margin-left: 400px;}"/>
            </ui>
            ]]
                errorui=simUI.create(warning)

        print("Target Torque Unreachable. Bolts were torqued to "..torque.." N-m")

    end


    -- Error is thrown when the input nominal pipe size is not found in database. A pop-up window appears to notify user and error is
    -- logged and the simulation is suspended.
    if typeError == "database" then
```

Simulations Demo

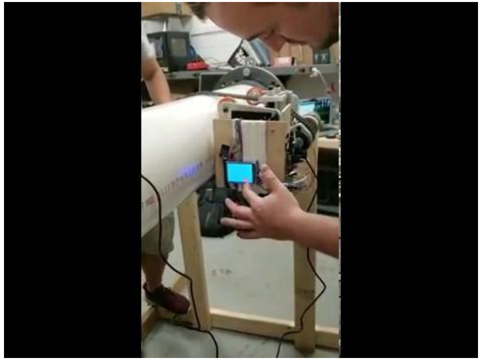
Validating Performance

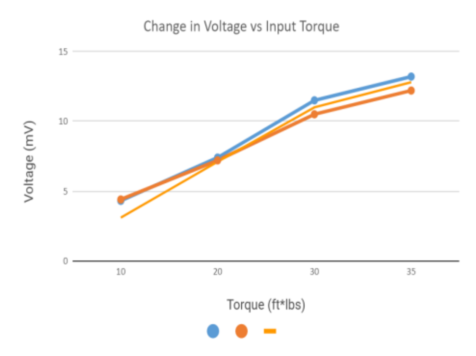
What we set out to do:

- ☐ Reach torque up to 40 ft-lbs.
  - Motor achieves 40 ft-lbs ✓
  - Strain gage function ✓
  - Torque on Nut ☐
- ☐ Rotate around surface of pipe
  - Move around Pipe ✓
  - Follow ASME Sequence ☐
- ☐ Able to align to bolt
- ☐ Interface socket with nut
  - Articulation slow lead ✓
  - Socket slow rotation ☐
- ✓ Software foundation
- ☐ Adapt to different pipe sizes

Results

- Circumnavigate Pipe
- Subassembly Integration
- Position Testing
- Strain Gauge Functionality





What we learned – Mechanical

Successes

- Designing functional mechanical systems
- How to be an effective team
- Working with multiple disciplines
- Problems Solving
- Understanding the customer's requirements
- Getting a product completed within budget

Challenges

- Number of iterations to succeed
- Working with large team
- Miscommunications
- Not identifying unexpected problems
- Failing to communicate with the customer
- Cuts made in designs to work within budget

What we learned - Electrical & Computer

Success

- Gaining hands on experience
- Learning topics outside of our discipline and how we can incorporate them into a design
- Dedicated team for project management
- Working with a diverse team added value to typical senior design experience

What we wish we could have done

- Have all parts ordered at the beginning of the semester
- Integration testing before ordering printed circuit board
- Had more time with the COBOT for enhanced testing procedures

What we learned – Computer Science

Successes

- Successful incorporation and manipulation of CAD designs from the MEs into V-REP
- Learned how to work with multiple disciplines on projects
- Learned concepts related to other disciplines to work on the project
- We learned how to apply the V-REP API and aspects of Lua programming.

Challenges

- V-REP script editor lacks any helpful tools for debugging, so we were forced to experiment. (We had to take out our print()
- V-REP force/torque sensor failed to give accurate data, so we created an algorithm.
- Features take longer to implement than we first imagined.

What we learned – Industrial

Successes

- Role of I.E.’s in large engineering teams
- Project Management
- Organization
- Effective Communication

Challenges

- Came in halfway through project
- Pushing back deadlines
- Establishing realistic objectives with the customer

Next Semester Advice/Ideas

Mechanical Engineering

- Secure bolt side
- Vertical case implementation
- One person installation procedure
- Brushless motors

Electrical & Computer Engineering

- Absolute encoder
- Camera for bolt alignment

Computer Science

- Implement proximity sensor
- Create application for wireless control
- VR simulation for training
- Modified UI
- Camera View

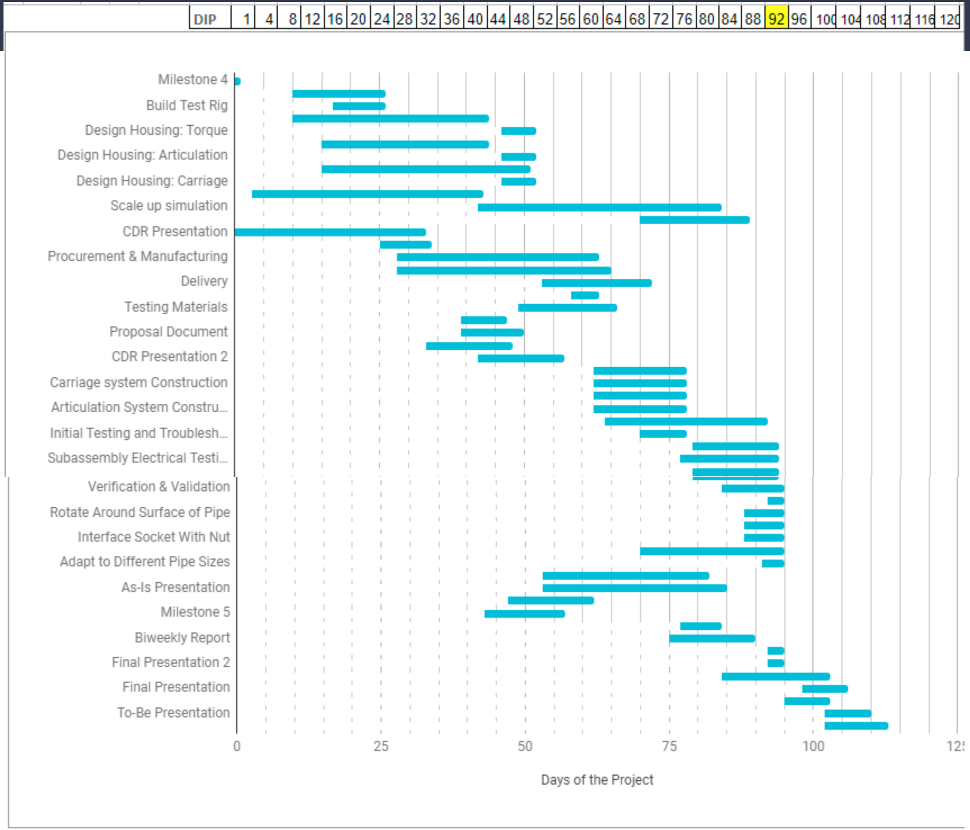
Industrial Engineering

- Develop one person installation procedure
- Safety analysis

Budget

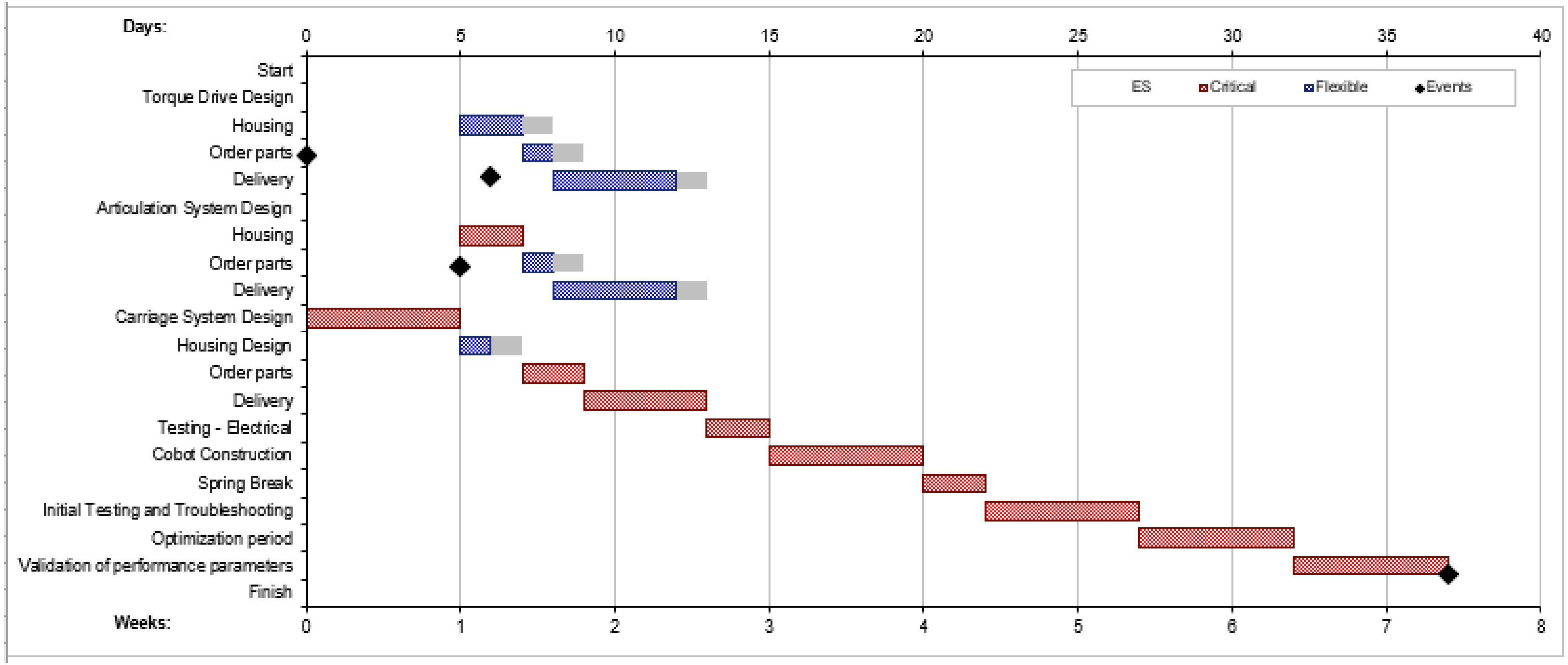
Supplier	Amount
Adafruit	$39.26
Amazon	$208.27
Arduino	$36.44
Arrow	$52.28
CPO Milwaukee	$208.00
DigiKey	$235.31
JLCPCB	$23.68
McMaster	$457.77
Mouser	$9.37
Omega	$72.00
PCBWay	$128.00
Skycraft	$38.00
US Digital	$89.77
Total	**$1,628.15**

Schedule

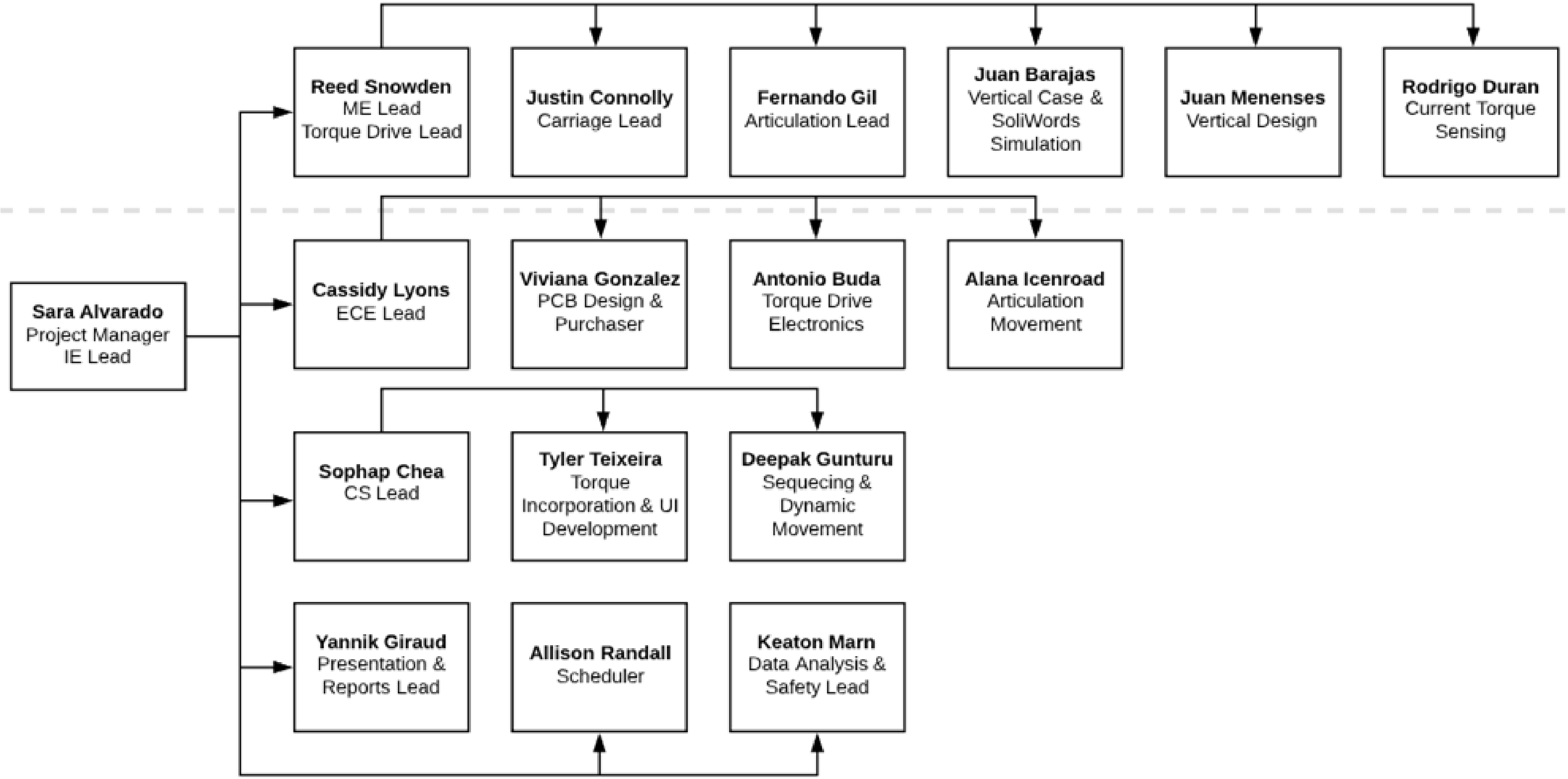


WBS	TASK NAME	START DATE	END DATE	START ON DAY*	DURATION* (WORK DAYS)	CHAMPION	PERCENT COMPLETE
1	**Milestone 4**	1/7	1/7	0	1	ME Team	100%
1	**Design Finalization**	1/17	2/1	10	16	ME Team	100%
1.1	Build Test Rig	1/24	2/1	17	9	ME Team	100%
1.2	Torque Drive Design	1/17	2/19	10	34	Reed	100%
1.3	Design Housing: Torque	2/22	2/27	46	6	Reed	100%
1.4	Articulation System Design	1/22	2/19	15	29	Fernando	100%
1.5	Design Housing: Articulation	2/22	2/27	46	6	Fernando	100%
1.6	Carriage System Design	1/22	2/26	15	36	Justin	100%
1.7	Design Housing: Carriage	2/22	2/27	46	6	Justin	100%
2	**Cobot Simulation**	1/10	4/15	3	96	CS Team	100%
2.1	Finish base case simulation	1/10	2/18	3	40	CS Team	100%
2.2	Scale up simulation	2/18	3/31	42	42	CS Team	100%
2.3	Test simulation	3/18	4/5	70	19	CS Team	100%
3	**CDR Presentation**	1/7	2/8	0	33	ECE Team	100%
4	**Biweekly Report**	2/1	2/9	25	9	IE Team	100%
5	**Procurement & Manufacturing**	2/4	3/10	28	35	ME Team	100%
5.1	Purchasing Material	2/4	3/12	28	37	Viviana	100%
5.2	Delivery	3/1	3/19	53	19	Vendor	100%
5.3	Housing Machining	3/6	3/10	58	5	ME Team	100%
5.4	Testing Materials	2/25	3/13	49	17	ME & ECE	100%
6	**Proposal Presentation**	2/15	2/22	39	8	IE Team	100%
7	**Proposal Document**	2/15	2/25	39	11	IE Team	100%
8	**Biweekly Report**	2/9	2/23	33	15	IE Team	100%
9	**CDR Presentation 2**	2/18	3/4	42	15	CS Team	100%
10	**Cobot Construction**	3/10	3/25	62	16	ME Team	100%
10.1	Carriage system Construction	3/10	3/25	62	16	ME TEam	100%
10.2	Torque Drive System Construction	3/10	3/25	62	16	ME Team	100%
10.3	Articulation System Construction	3/10	3/25	62	16	ME Team	100%
11	**Cobot Assembly Testing**	3/12	4/8	64	28	ALL	71%
11.1	Initial Testing and Troubleshooting	3/18	3/25	70	8	ME Team	75%
11.2	Strain Gauge Testing	3/27	4/10	79	15	ME & ECE	50%
11.3	Subassembly Electrical Testing	3/25	4/10	77	17	ECE Team	85%
11.4	Cobot assembly Electrical Testing	3/27	4/10	79	15	ME & ECE	75%
12	**Verification & Validation**	4/1	4/11	84	11	ALL	17%
12.1	Reach 40 ft-lbs	4/9	4/11	92	3	ME & ECE	0%
12.2	Rotate Around Surface of Pipe	4/5	4/11	88	7	ME & ECE	25%
12.3	Align With Bolt	4/5	4/11	88	7	ME & ECE	0%
12.4	Interface Socket With Nut	4/5	4/11	88	7	ME & ECE	0%
12.5	Software Foundation	3/18	4/11	70	25	ALL	50%
12.6	Adapt to Different Pipe Sizes	4/8	4/11	91	4	ME & ECE	25%
13	**As-Is Report**	3/1	3/29	53	29	IE Team	100%
14	**As-Is Presentation**	3/1	4/1	53	32	IE Team	100%
15	**Biweekly Report**	2/23	3/9	47	15	IE Team	100%
16	**Milestone 5**	2/19	3/4	43	14	ME Team	100%
17	**Mid-Term Demo**	3/25	3/31	77	7	CS & ECE	100%
18	**Conference Paper and Committee Form**	1/7	4/19	0	103	ECE Team	0%
19	**Biweekly Report**	3/23	4/6	75	15	IE Team	100%
20	**Final Presentation**	4/9	4/11	92	3	CS Team	25%
21	**Final Presentation 2**	4/9	4/11	92	3	ALL	25%
22	**To-Be Report**	4/1	4/19	84	19	IE Team	0%
23	**Final Presentation**	4/15	4/22	98	8	IE Team	0%
24	**Milestone 6**	4/12	4/19	95	8	ME Team	0%
26	**To-Be Presentation**	4/19	4/26	102	8	IE Team	0%
27	**Final Poster**	4/19	4/29	102	11	IE Team	0%

Critical Path



Roles + Responsibilities



A BIG Thank You

Siemens Sponsors

Gerald J. Feller, PhD

Paul Zombo

Matt Johnson

Daniel Ryan

Don Stabile

Advisers

Kurt Stresau & Bonnie Marini, PhD (ME)

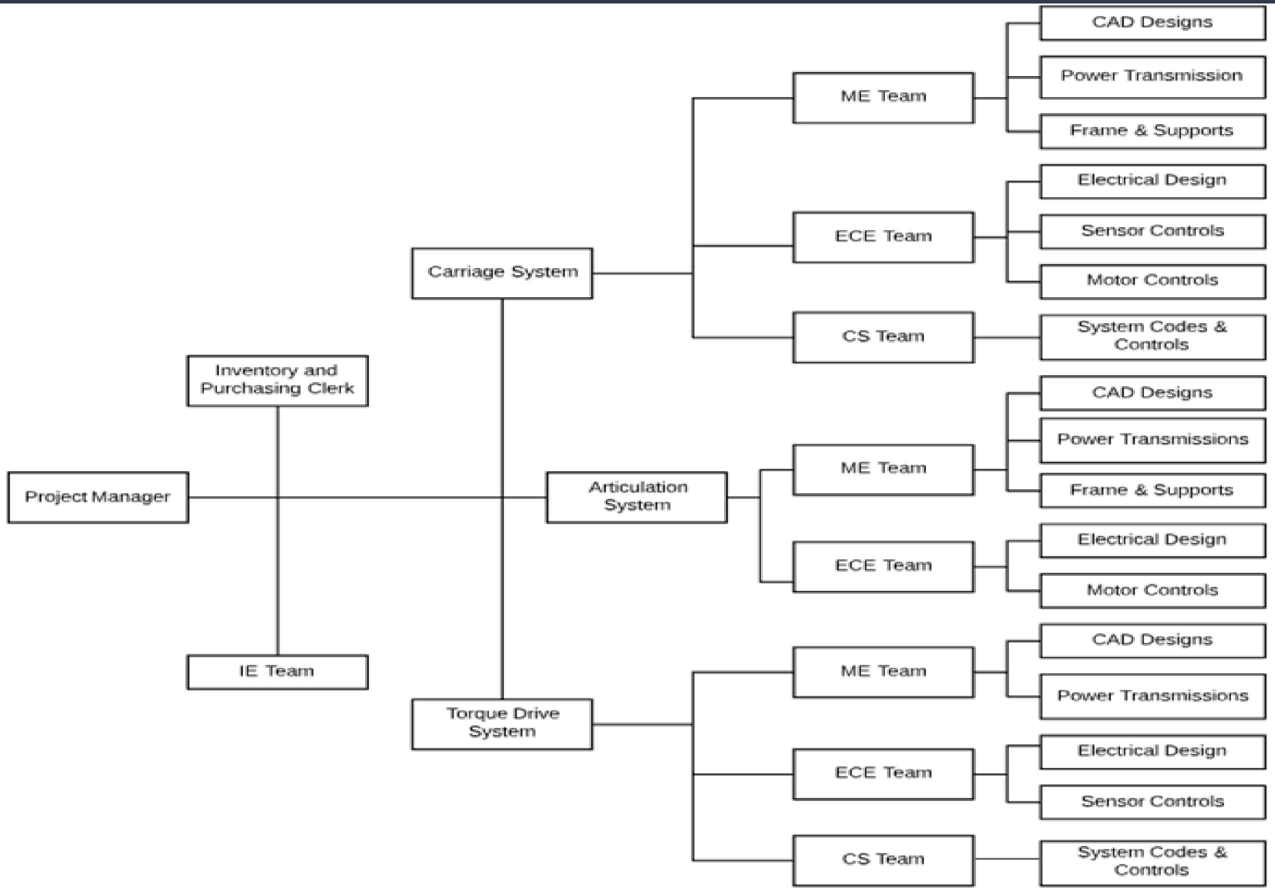
Mike Conroy & Mark Heinrich, PhD (CS)

Samuel Richie, PhD (ECE)

Luis Rabelo, PhD (IE)

Demonstration

Organizational Chart



Division of Work (ME)

Reed Snowden (Lead)	Justin Connolly	Fernando Gil
• Torque drive system design, construction, testing, analysis • Machining • Strain gage testing • Test rig construction • Wheatstone bridge construction and testing	• Carriage system design, construction, testing, and analysis • Housing machining • CAD modeling	• Articulation system design, construction, testing and analysis • Test rig construction • 3D parts design and printing • ME Basic Coding for testing
Juan Barajas	**Juan Meneses**	**Rodrigo Duran**
• Vertical concept CAD modeling • Scale up CAD model • Articulation system CAD model • Milestones • Weekly Memos • Test rig construction	• Test rig construction • Vertical concept Design and CAD modeling • Milestones • Carriage design	• Wheatstone bridge concept and testing • Milestones • Carriage design • Rotary/Shaft Encoder

Division of Work (ECE)

Cassidy Lyon (lead)	Viviana Gonzalez Pascual	Alana Icenroad	Antonio Buda
• Carriage system electronic components • Encoder code and testing • Wiring • Electronic component integration and testing	• PCB design, implementation, and testing • Inventory and purchasing clerk • Testing of electronic components • Power supply design and implementation • Develop schematics	• Articulation subsystem movement • Stepper motor code	• Integrating electronics for torque drive subsystem • Wheatstone bridge development

Division of Work (CS)

Sopheap Chea (lead)	Tyler Teixeira	Deepak Gunturu
• Synergize code segments into one • CAD incorporations • Internal database • Proximity sensor • Error Handling • Vertical design simulation • Scale-up simulations • Data/Crash log	• Create UI for user Torque equation on to nut/bolt • Proto-Cobot simulation • Estimated task time • Debugger	• Relate input data to find appropriate sequence • Move Cobot around pipe in sequence • Two tool simulation

Division of Work (IE)

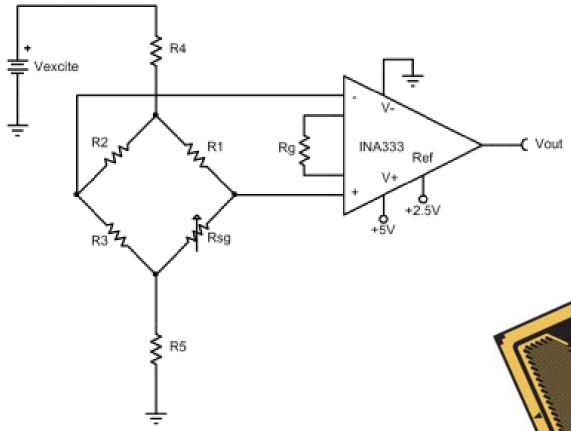
Sara Alvarado (lead)	Yannick Giraud	Allison Randall	Keaton Marn
• Project management • Enable communication between disciplines • Update CPM chart • Weekly report to Siemens • Develop FMEA • Lead weekly meetings • Time studies on manual flange assembly	• Presentation lead • Formatting of reports for IE and ME reports • IE assignment lead	• Schedule meetings and organize meeting rooms • Create Gantt Chart • Create CPM chart • Update Gantt chart and CPM chart	• Data analysis • Safety/Ergonomic lead • Assess current process non-conformities and injuries • Analize time study date on manual flange assembly

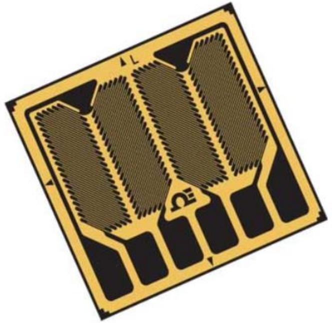
Tony (EE)

Main responsibility: Integrating electronics for the torque drive subsystem.

Learning experiences: How to use a strain gauge to measure motor torque an amplify the signal to be read by an arduino.

Proper components selection for specific purposes.



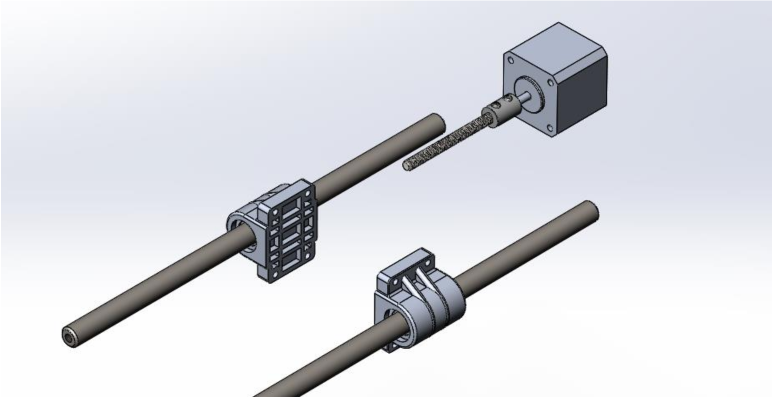


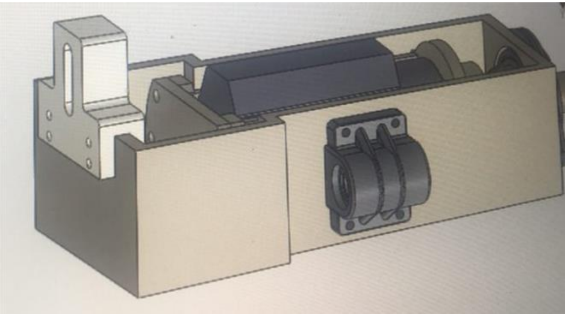
Alana (CpE)

Main Role: Articulation Subsystem movement

Learning Experience: Gaining knowledge about how different motors work - something I would not have learned about without involvement in this project

Defining a constant distance regardless of speed makes my code reliable

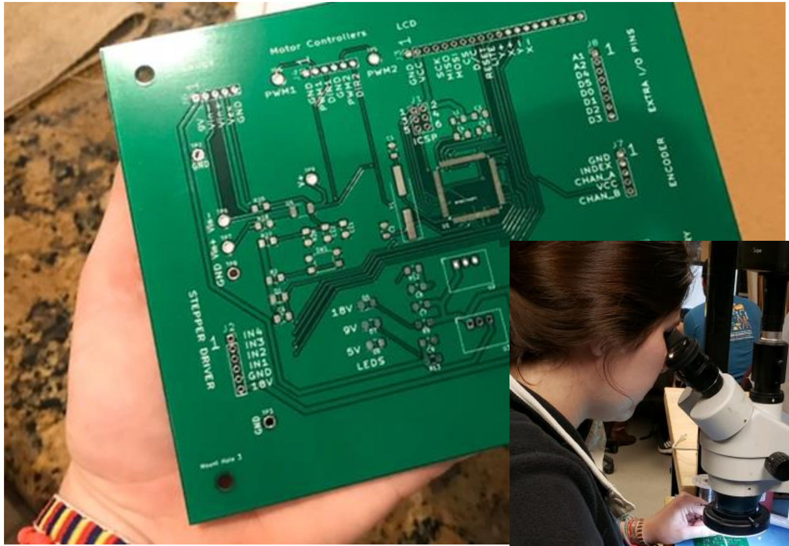




Viviana (ECE)

Main role:

- Team’s Purchasing Clerk
- Concept Design for PCB and Power Circuits
- Created Schematic
- Layed out MCU PC Board
- Solder all components on MCU Board
- Test PCB functionality



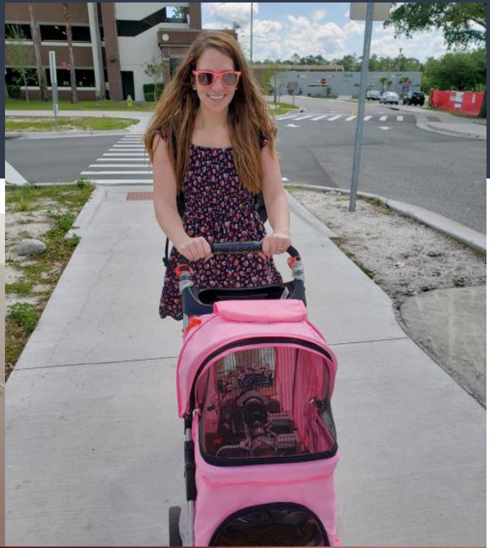
Cassidy (CpE)

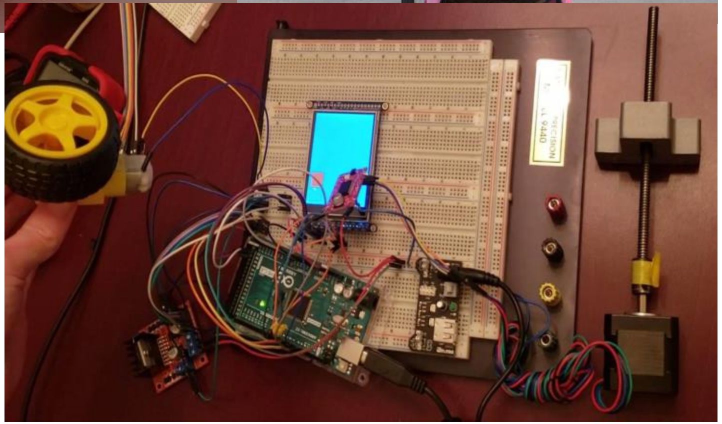
Main role: Carriage Subsystem (Interface with Encoder)

Most exciting event:

Making each part move and work together - not just carriage motor specific

System as a whole is far more impressive than any one component





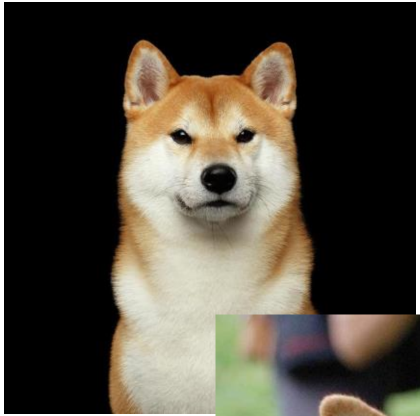
Rodrigo (M.E.)

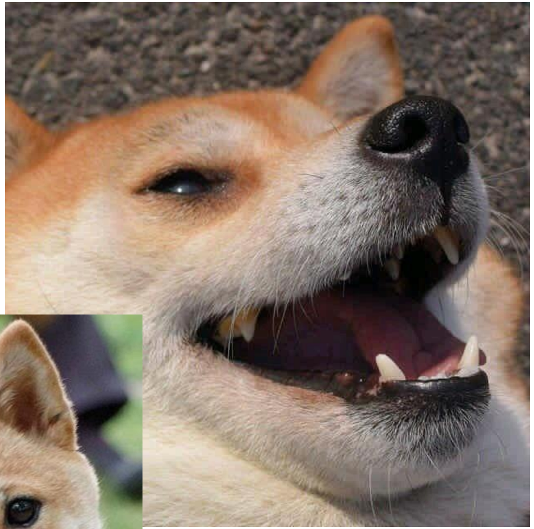
Main Roles:

- Concept Design
- Rotary/Shaft Encoder Research
- Current Torque Sensing Technologies
- Open Loop: Current controlled
- Closed Loop: Strain Gauge
- Changed Legal Name to “Juan Duran”

Learning Experiences:

- Teamwork and Responsibilities
- Presentation Skills
- Engineering Reports

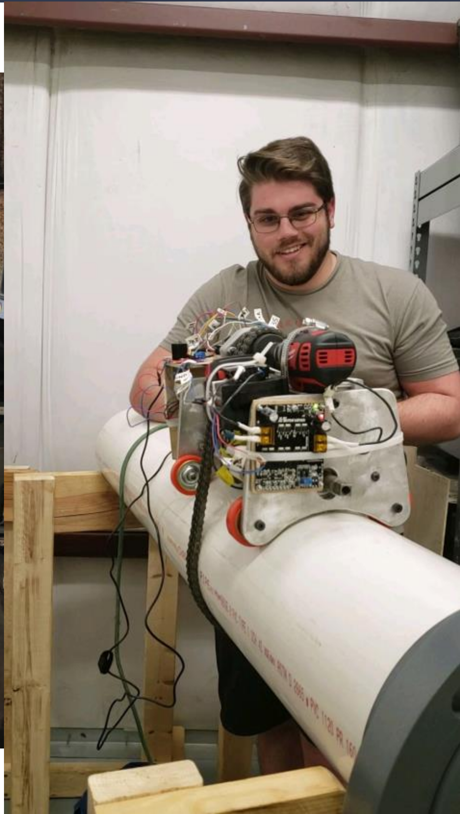




Justin (M.E.)

Main Roles
- Carriage and Pipe Traversal System Lead Designer
- Static & Dynamic Calculations
- Subsystem Integration
- Perform and Catalog CAD Updates

Greatest Learning Experiences
- Working with and (at times) managing a large multidisciplinary team to achieve subsystem goals
- Design for manufactuability, assembly, cost
- Importance of developing good requirements
- Importance of excellent communication among team members and customer



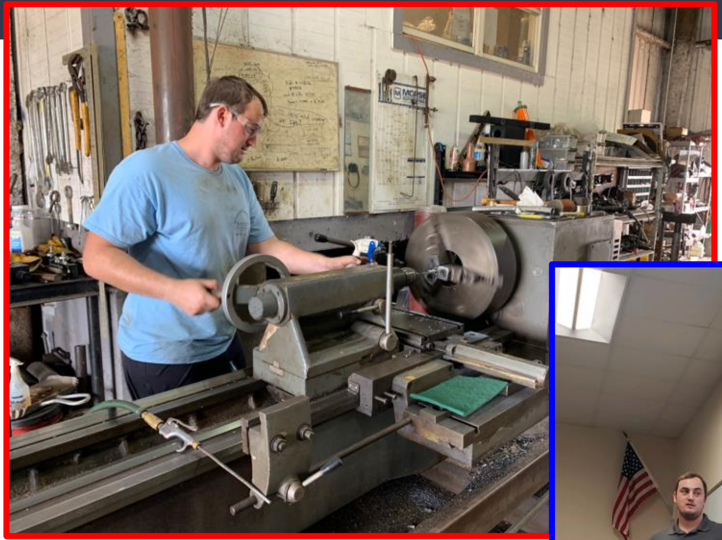
Reed (M.E.)

Main Roles

- Torque Drive System and concept design lead
- Static failure calculations
- Manufacturing lead/ Coordinator
- Perform CAD of Torque system and integration with other systems

Greatest Learning Experiences

- Design for Manufacturing, Assembly, and Cost
- The complexities of electro-mechanical systems
- Scheduling and the importance of goal setting
- Having organised dialogue with other teams



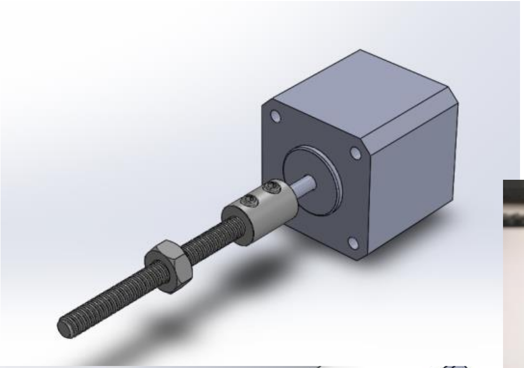
Fernando Gil (M.E)

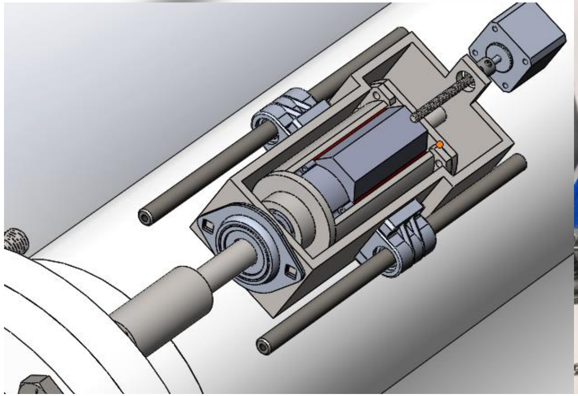
Main Roles

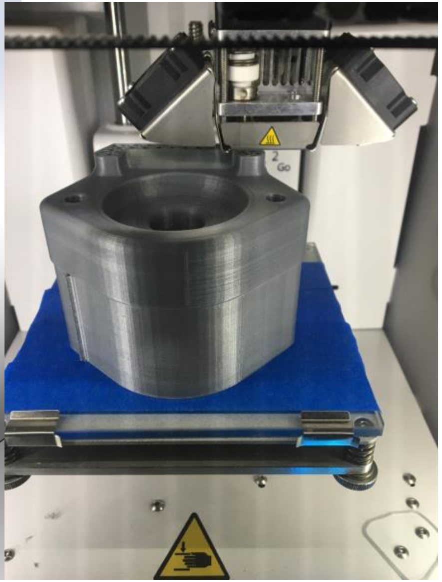
- Articulation System Development Lead
- Overall Subsystem Integration.
- ME Organizer

Greatest Learning experiences

- Designing a functional and adaptable system that integrated other subsystems.
- Importance of communication between the two subsystem teams.
- How to help other teams even if they are from other sub-systems ranging from debugging, electrical wiring, design reviews, and brainstorming.







Juan Barajas (M.E)

Main Roles: SolidWorks animations, vertical case and articulation models. Scale up of pipe and flanges.

Learning experiences: Hands on experience with removing material from parts to properly attach them together. Just because it was modeled in a software doesn't mean it is going to be perfect.

Collaborated with team members from different majors to help model vertical case concepts and scale up models of pipes and flanges.

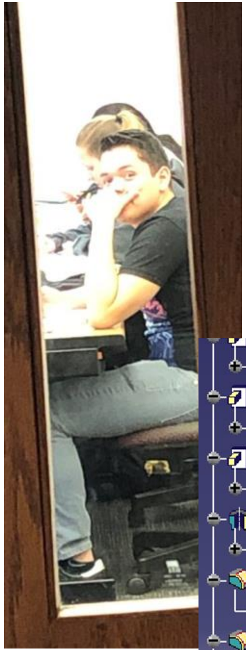
Juan Meneses (M.E.)

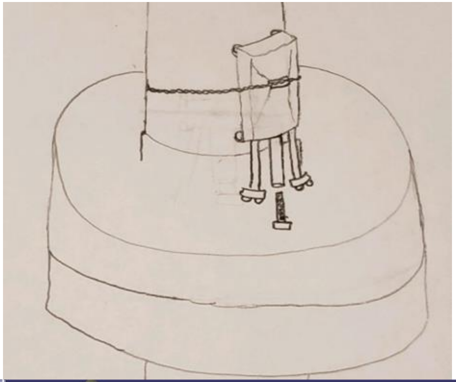
Main Role:

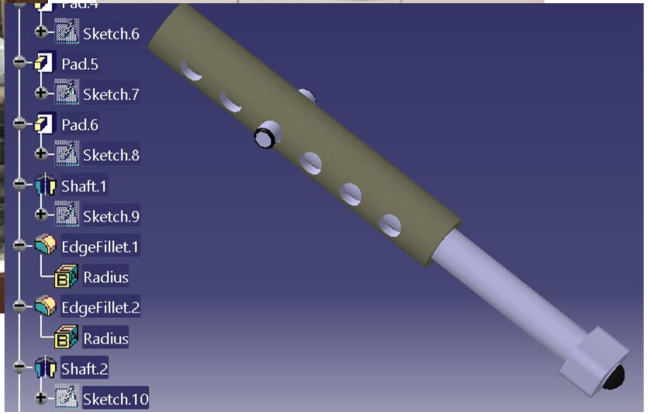
- Design of Vertical Case Models & Integration to Cobot Model
- Ensure ME Milestones where delivered

Learning Experience:

- Integrate CAD models into assemblies
- Work with other disciplines to help deliver deadlines
- Researching and Understanding Trademarks







Sopheap (CS)

Main Roles: CS team manager, CAD to V-REP incorporation, scale ups and vertical simulation.

Greatest learning experiences:

Learned to be in a role where I am not only responsible for my own work, but also to ensure the work of the others within my team gets done, so that our team as a whole is successful.

Acting as the “middle man” between my team and the other interdisciplinary teams, I was able to learned a bit about hardware and the design process from the others.

Deepak (CS)

Main role: Responsible for the dynamic movement of the Cobot simulation in the ASME star pattern for flanges with different number of bolts and simulating the two-Cobot design for optimized functionality

Greatest learning experiences:

Working on an interdisciplinary project, I was able to learn about how to work and coordinate with people from other disciplines and develop a product. Learning robotic simulation with V-REP and getting the Cobot designs to work in a step-by-step manner was a great learning experience for me as well.

Tyler (CS)

Primary Responsibilities: Creating a time estimation algorithm, calculating applied torque, implementing a user interface, and completion of the horizontal simulation.

Additional Contributions: Helped problem solve and debug in other aspects of the project. Offered alternative solutions and ideas when needed.

Difficulties Experienced: V-REP script editor lacks any helpful tools for debugging, V-REP joints failed to give accurate data for torque, lack of experience with the Lua programming language and V-REP API.

Sara (I.E.)

Role: Project Manager, Industrial Team lead

Contribution:

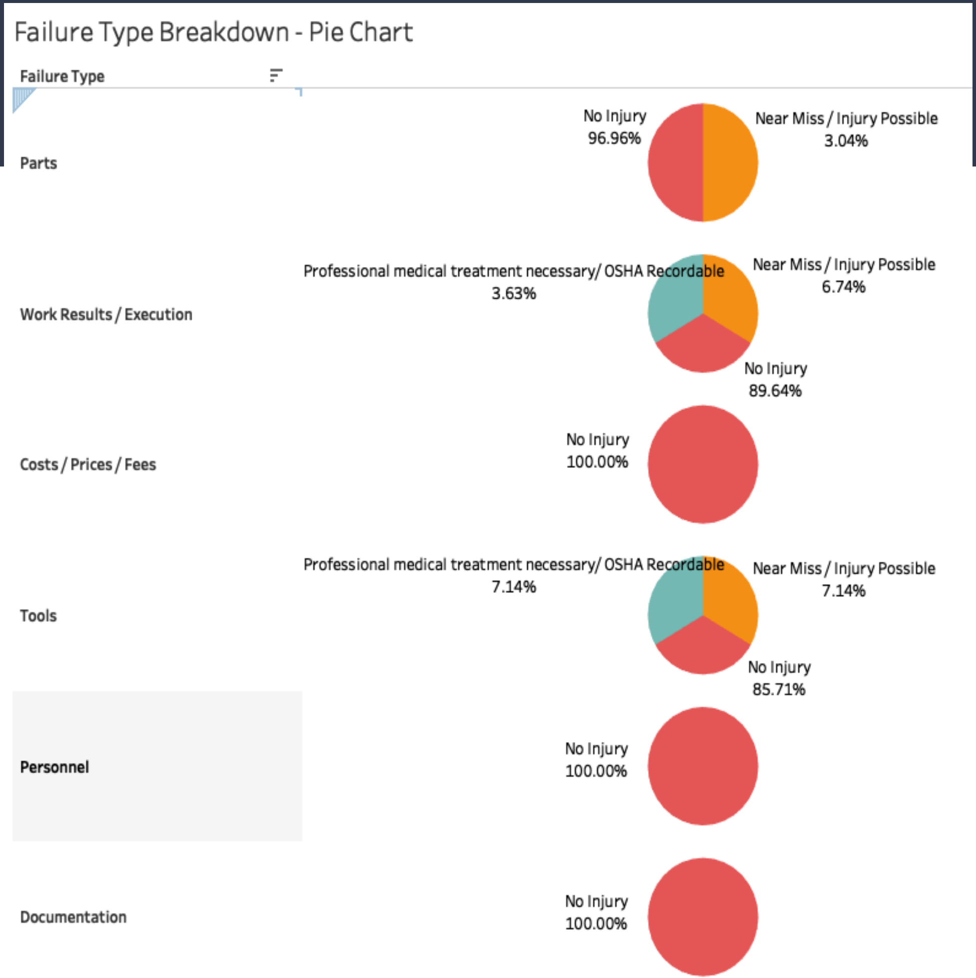
- Value Added
  - Leadership
  - Integration
  - Organization
  - Weekly Updates to Siemens
- Value Gained

Keaton (I.E.)

Main Role: Data Analyst and Safety/Ergonomic Lead - utilized programs such as Tableau and Simio to manipulate and interpret data and system analysis tools to understand the flow and potential failures within the process.

Greatest Learning Experience:

Gaining a better understanding of the complex flow of information and numerous relationship necessary within an interdisciplinary team to complete the project. Getting to partake in each stage of the process of developing the Cobot and observing how, at every step, each discipline and group member was crucial to the success of the project.



Allison (I.E)

Critical Path Method

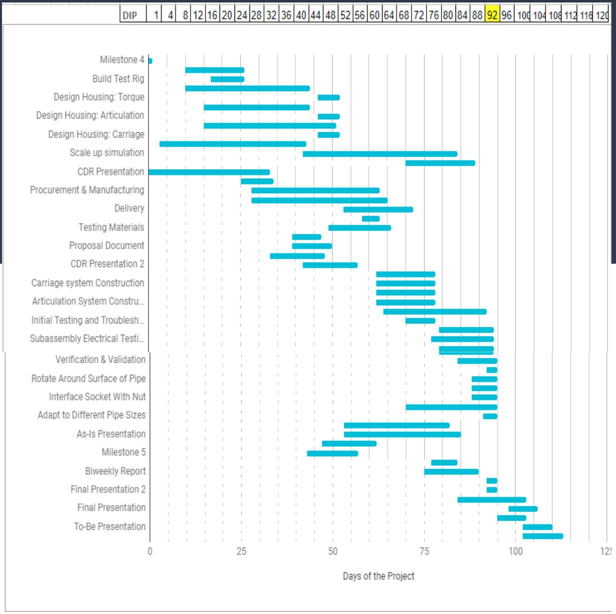
HELP

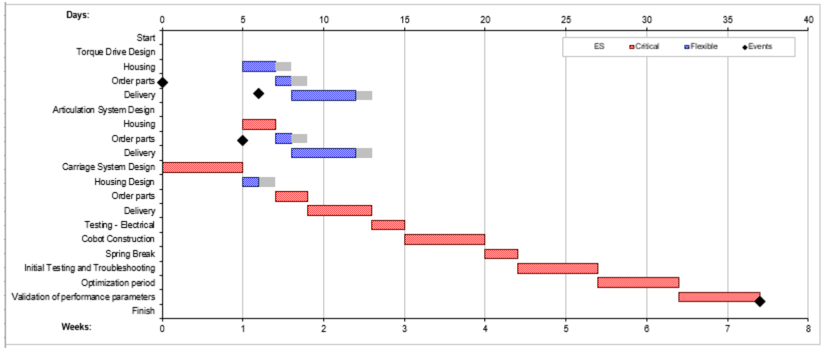
Start Date		Days to Completion
2/19/2019		37.00
Finish Date		
4/11/2019	<- assumes weekdays	

Times (in Days)

Time Distribution: Triangular

ID	Task Name	Predecessors (Enter one ID per cell)			O (min)	M (most	P (max)
1	**Start**						
2	**Torque Drive Design**	10			0	0	0
3	**Housing**	2			2	2	2
4	**Order parts**	3			1	1	1
5	**Delivery**	4			2	5	5
6	**Articulation System Design**	10			0	0	0
7	**Housing**	6			2	2	2
8	**Order parts**	7			1	1	1
9	**Delivery**	8			2	5	5
10	**Carriage System Design**	1			5	5	5
11	**Housing Design**	10			1	1	1
12	**Order parts**	11	7	2	2	2	2
13	**Delivery**	12			2	5	5
14	**Testing - Electrical**	5	9	13	2	2	2
15	**Cobot Construction**	14			5	5	5
16	**Spring Break**	15			2	2	2
17	**Initial Testing and Troubleshooting**	16			5	5	5
18	**Optimization period**	17			5	5	5
19	**Validation of performance parameter**	18			5	5	5
20	**Finish**	19					





WBS	TASK NAME	START DATE	END DATE	START ON DAY*	DURATION* (WORK DAYS)	CHAMPION	PERCENT COMPLETE
1	**Milestone 4**	1/7	1/7	0	1	ME Team	[illegible]
1	**Design Finalization**	1/17	2/1	10	16	ME Team	100%
1.1	Build Test Rig	1/24	2/1	17	9	ME Team	100%
1.2	Torque Drive Design	1/17	2/19	10	34	Reed	100%
1.3	Design Housing: Torque	2/22	2/27	46	6	Reed	100%
1.4	Articulation System Design	1/22	2/19	15	29	Fernando	100%
1.5	Design Housing: Articulation	2/22	2/27	46	6	Fernando	100%
1.6	Carriage System Design	1/22	2/26	15	36	Justin	100%
1.7	Design Housing: Carriage	2/22	2/27	46	6	Justin	100%
2	**Cobot Simulation**	1/10	4/15	3	96	CS Team	100%
2.1	Finish base case simulation	1/10	2/18	3	40	CS Team	100%
2.2	Scale up simulation	2/18	3/31	42	42	CS Team	100%
2.3	Test simulation	3/18	4/5	70	19	CS Team	100%
3	**CDR Presentation**	1/7	2/8	0	33	ECE Team	100%
4	**Biweekly Report**	2/1	2/9	25	9	IE Team	100%
5	**Procurement & Manufacturing**	2/4	3/10	28	35	ME Team	100%
5.1	Purchasing Material	2/4	3/12	28	37	Viviana	100%
5.2	Delivery	3/1	3/19	53	19	Vendor	100%
5.3	Housing Machining	3/6	3/10	58	5	ME Team	100%
5.4	Testing Materials	2/25	3/13	49	17	ME & ECE	100%
6	**Proposal Presentation**	2/15	2/22	39	8	IE Team	100%
7	**Proposal Document**	2/15	2/25	39	11	IE Team	100%
8	**Biweekly Report**	2/9	2/23	33	15	IE Team	100%
9	**CDR Presentation 2**	2/18	3/4	42	15	CS Team	100%
10	**Cobot Construction**	3/10	3/25	62	16	ME Team	100%
10.1	Carriage system Construction	3/10	3/25	62	16	ME TEam	100%
10.2	Torque Drive System Construction	3/10	3/25	62	16	ME Team	100%
10.3	Articulation System Construction	3/10	3/25	62	16	ME Team	100%
11	**Cobot Assembly Testing**	3/12	4/8	64	28	ALL	71%
11.1	Initial Testing and Troubleshooting	3/18	3/25	70	8	ME Team	75%
11.2	Strain Gauge Testing	3/27	4/10	79	15	ME & ECE	50%
11.3	Subassembly Electrical Testing	3/25	4/10	77	17	ECE Team	85%
11.4	Cobot assembly Electrical Testing	3/27	4/10	79	15	ME & ECE	75%
12	**Verification & Validation**	4/1	4/11	84	11	ALL	17%
12.1	Reach 40 ft-lbs	4/9	4/11	92	3	ME & ECE	0%
12.2	Rotate Around Surface of Pipe	4/5	4/11	88	7	ME & ECE	25%
12.3	Align With Bolt	4/5	4/11	88	7	ME & ECE	0%
12.4	Interface Socket With Nut	4/5	4/11	88	7	ME & ECE	0%
12.5	Software Foundation	3/18	4/11	70	25	ALL	50%
12.6	Adapt to Different Pipe Sizes	4/8	4/11	91	4	ME & ECE	25%
13	**As-Is Report**	3/1	3/29	53	29	IE Team	100%
14	**As-Is Presentation**	3/1	4/1	53	32	IE Team	100%
15	**Biweekly Report**	2/23	3/9	47	15	IE Team	100%
16	**Milestone 5**	2/19	3/4	43	14	ME Team	100%
17	**Mid-Term Demo**	3/25	3/31	77	7	CS & ECE	100%
18	**Conference Paper and Committee Form**	1/7	4/19	0	103	ECE Team	0%
19	**Biweekly Report**	3/23	4/6	75	15	IE Team	100%
20	**Final Presentation**	4/9	4/11	92	3	CS Team	25%
21	**Final Presentation 2**	4/9	4/11	92	3	ALL	25%
22	**To-Be Report**	4/1	4/19	84	19	IE Team	0%
23	**Final Presentation**	4/15	4/22	98	8	IE Team	0%
24	**Milestone 6**	4/12	4/19	95	8	ME Team	0%
26	**To-Be Presentation**	4/19	4/26	102	8	IE Team	0%
27	**Final Poster**	4/19	4/29	102	11	IE Team	0%

Yannick (I.E.)

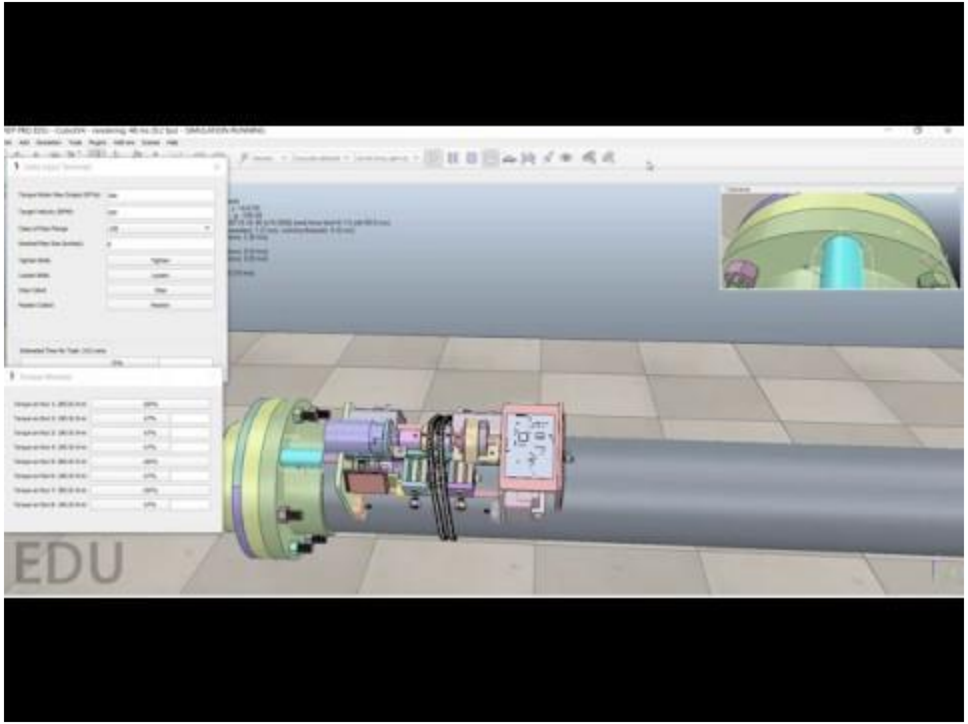
Main Role

- Presentation lead
- Formatting
- I.E. Assignments

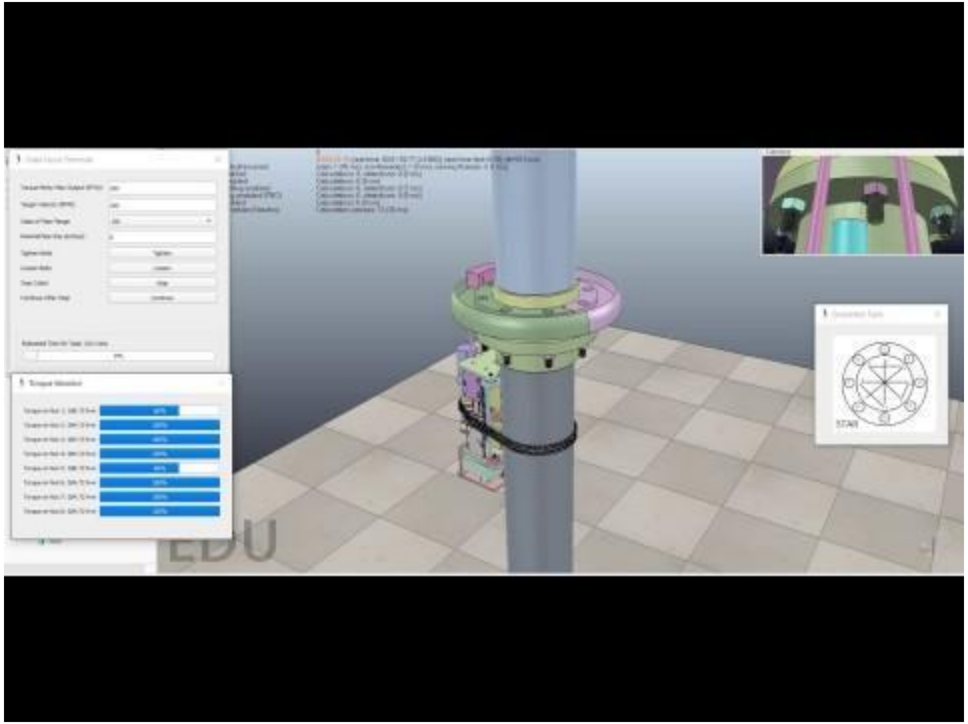
Greatest Learning Experience

- Understanding the core responsibilities of an I.E. or project manager
- Constant communication is key to a successful project

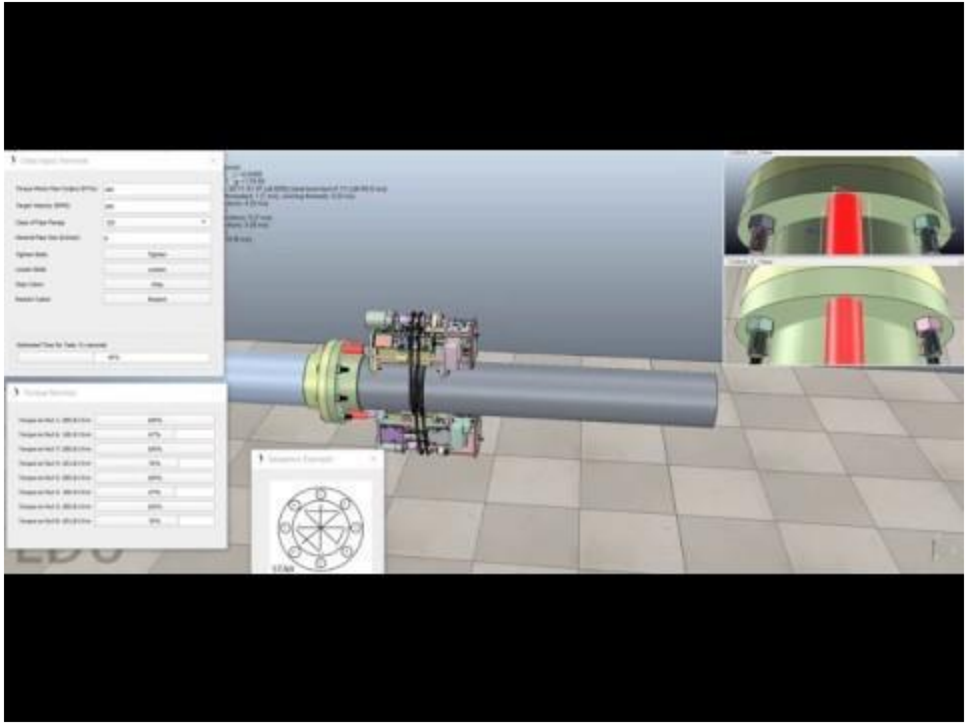
Proto-Cobot Simulation Video



Upward Design Simulation Video



2 Cobot Simulation Video



Block Diagram