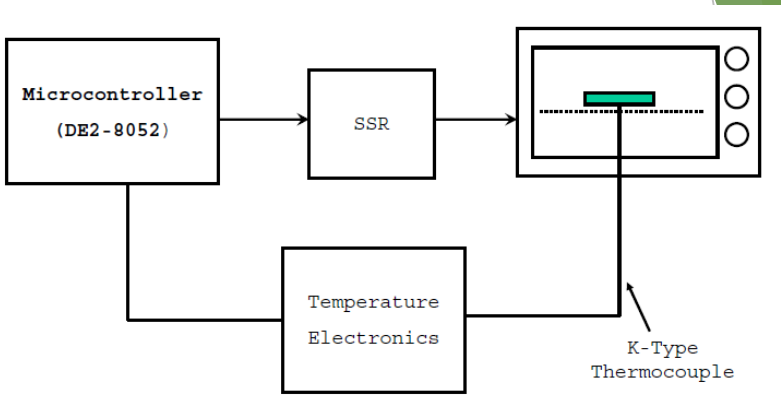
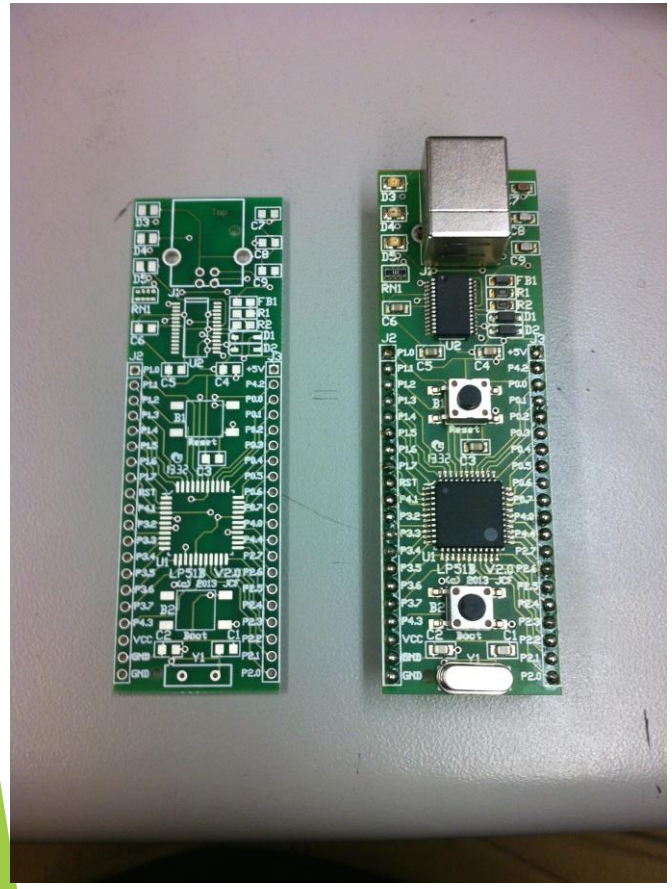
Engineering Design Portfolio

Glyn Han
Electrical Engineering Student
The University of British Columbia

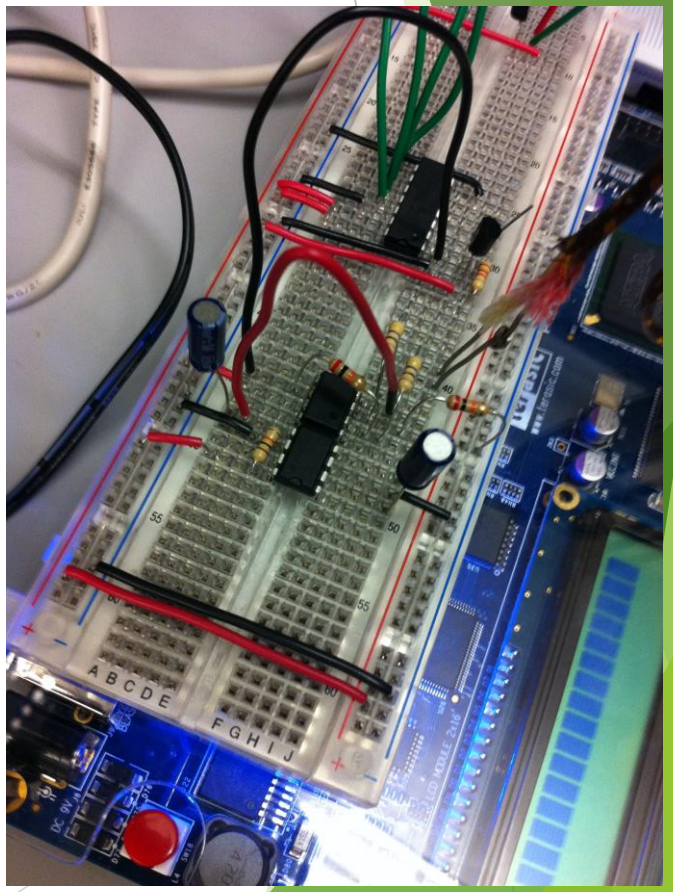
Table of Contents

Reflow Oven Controller



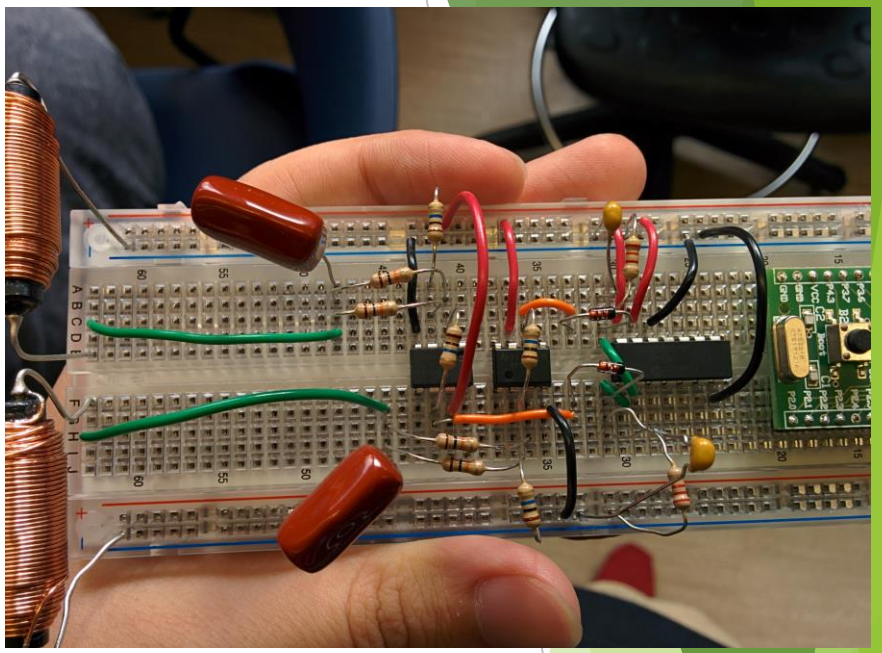


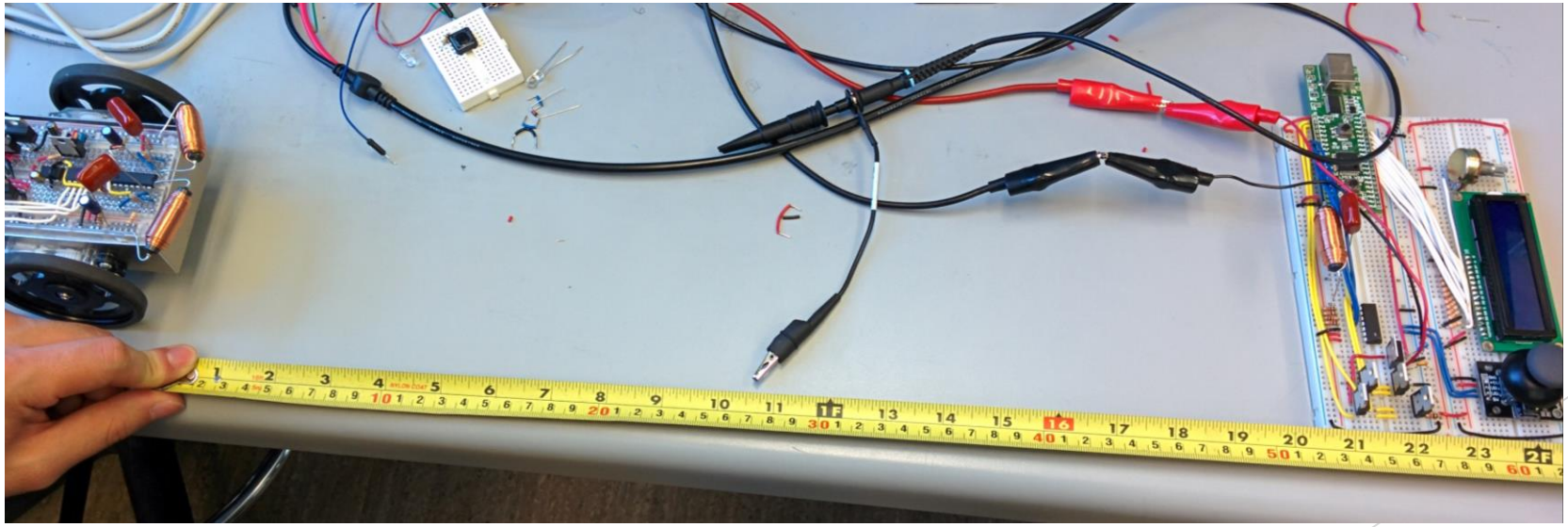
- Part of a 6 member team
- Concentrated work on hardware, and circuit prototyping
- Designed a circuit attached to a FPGA that controlled an ovens on and off function
- Reflow soldered lp51b microcontrollers



Electromagnetic Tether Robot

- Part of a 6 member team
- Majority of work done on circuitry and hardware prototyping
- Bent laser cut out of car body
- Utilized H bridge configuration for motor control
- Magnetic transmitters from ferrite inductors





Pong Game VHDL Hardware Design

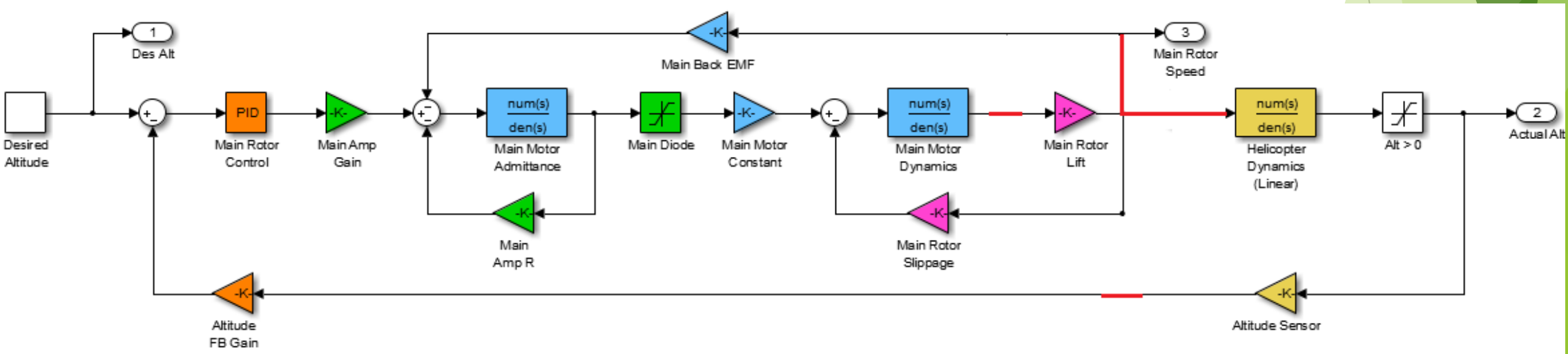
- Worked with a partner in designing a VGA port display Pong game
- Utilized FPGA for easy prototyping and troubleshooting
- Added additional challenge to the game that increases speed over time
- Sample code to the right

```
when initial =>
      plot <= '1';
      colour <= "111";--white
      y <= std_logic_vector(temp_y);
      x <= std_logic_vector(temp_x);
      temp_x := temp_x + 1;--continue at this state on clk until condition
      if (temp_x = 155) then
          temp_x := "00000101";--boundary on x
          if (temp_y = 115) then
              goalie1 := "0110110";--54
              forward1 := "0110110";
              goalie2 := "0110110";
              forward2 := "0110110";
              temp_y := "0000101";--5
              state := team1_g;
          else
              temp_y := "1110011";-- 120-5=115
          end if;
      end if;

 when team1_g =>
      plot <= '1';
      temp_x := "00000101";
      temp_y := temp_y + 1;-- continue until end of y=115
      if (temp_y >= goalie1 and temp_y <= goalie1 + 10 and temp_y < 115) then
          colour <= "001";
      elsif (temp_y >= 115) then
          if (sw(17) = '1') then
              if (goalie1 > 6) then
                  goalie1 := goalie1 - 1;
              end if;
          else                              --goalie going down
              if (goalie1 < 104) then    --115-10-1, top of goalie
                  goalie1 := goalie1 + 1;
              end if;
          end if;
          temp_y := "0000101";
          colour <= "111";
          state := team1_f;
      else
          colour <= "000";
      end if;
      y <= std_logic_vector(temp_y);
      x <= std_logic_vector(temp_x);
```

Apache Flight Control MATLAB/SIMULINK DESIGN

- My partner and I developed and designed a PID controller with a route error average of 7.08 offset from original flight course
- Adjusted the PID values capable of handling a marginally stable helicopter system modelled through Simulink
- Determined transfer functions, used root locus, and nyquist criterion to analyze stability positions of the model
- Simulated entire design and constantly working to decrease the error in the overshoot while keeping constant and consistent stability though its entire flight path



Apache Transfer Functions

- Closed loop transfer function of the entire system simplified

$$\textbf{Closed Loop TF:}\quad \textbf{CLsys} = \textbf{TF4} = \frac{KG}{1 + K * G * H} = \frac{\left(\frac{5.306}{s^4 + 31.33 * s^3 + 221.5 * s^2}\right)}{1 + \left(\frac{5.306}{s^4 + 31.33 * s^3 + 221.5 * s^2}\right)}$$

$$\textbf{CLsys} = \textbf{TF4} = \frac{5.30597014925}{5.30597014925 + (s^4 + 31.3286902287 * s^3 + 221.498901943 * s^2)}$$

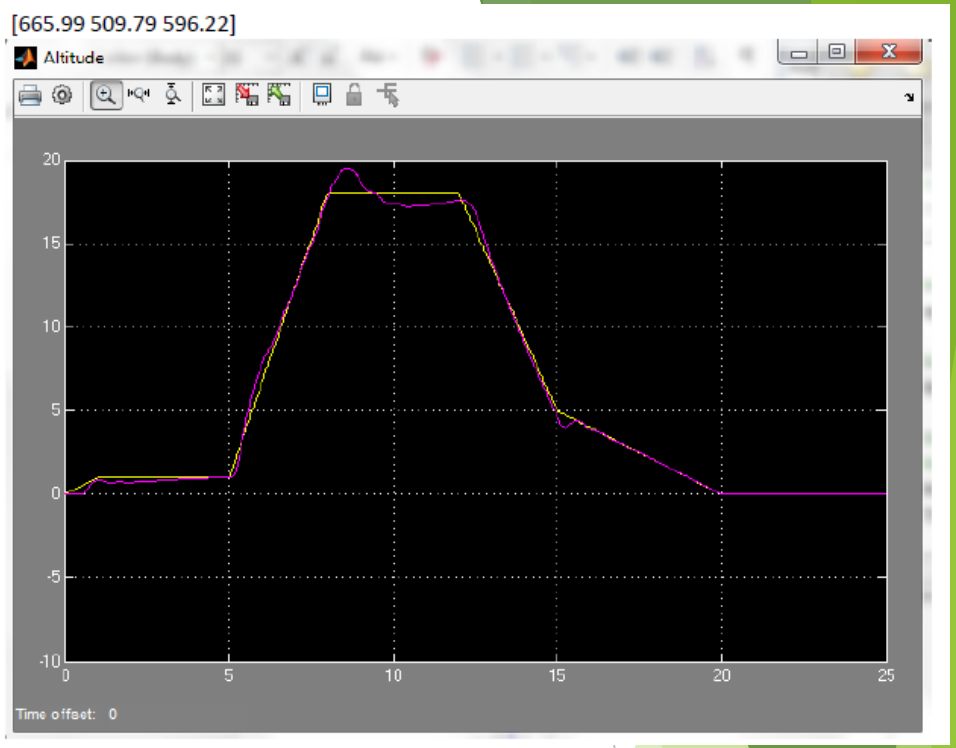
$$\textbf{CLsys} = \frac{5.31}{s^4 + 31.3 * s^3 + 222 * s^2 + 5.31}$$

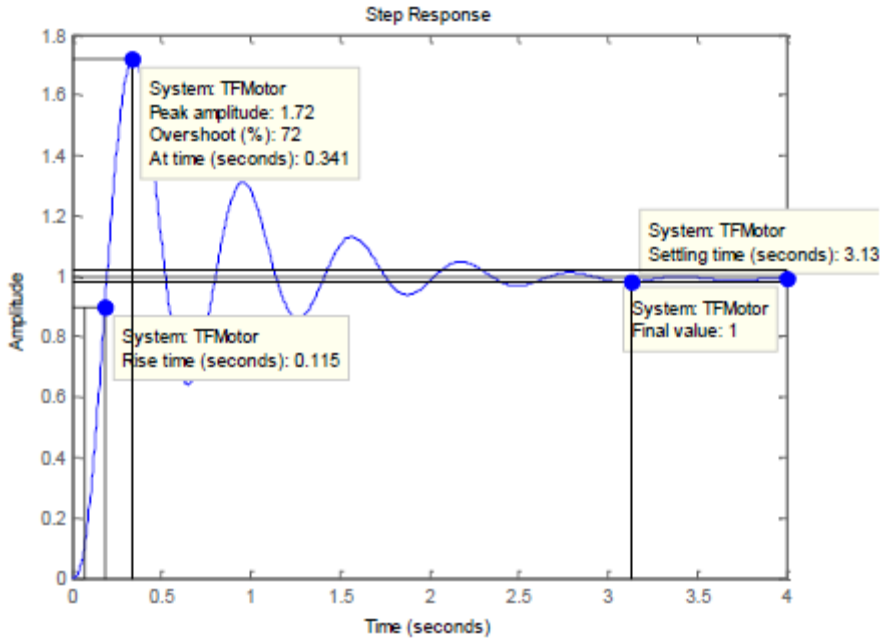
- Open loop transfer function determined

$$\textbf{Open Loop TF:}\quad \textbf{OLsys} = K * G * H = \frac{5.31}{s^4 + 31.3 * s^3 + 222 * s^2}$$

Apache Flight Simulations

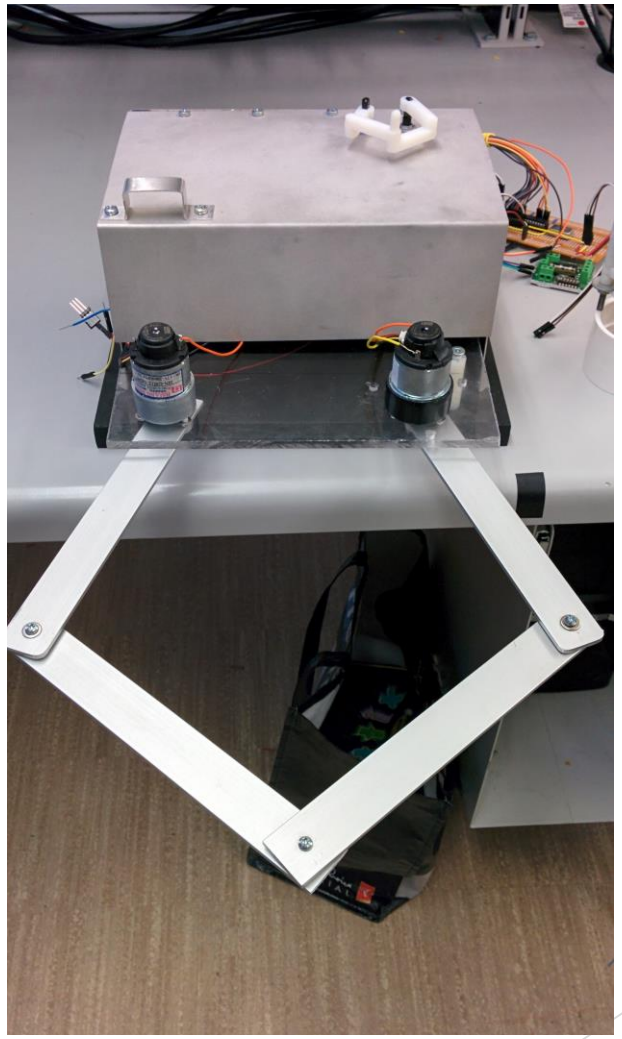
- Flight path given in the yellow graph and simulated flight path in the purple
- Below is the step response of the system using the calculated transfer functions

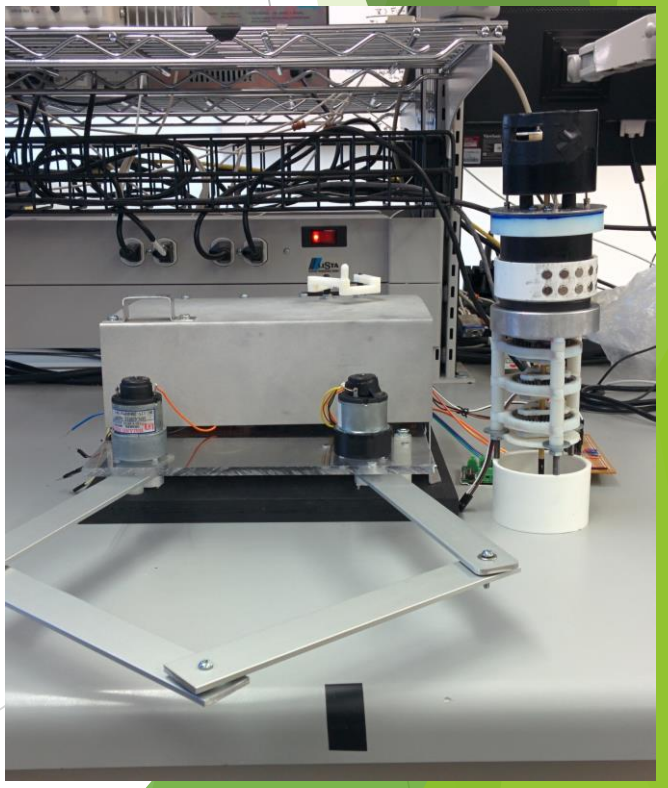




5-Bar Robotic Arm and Electric Motor

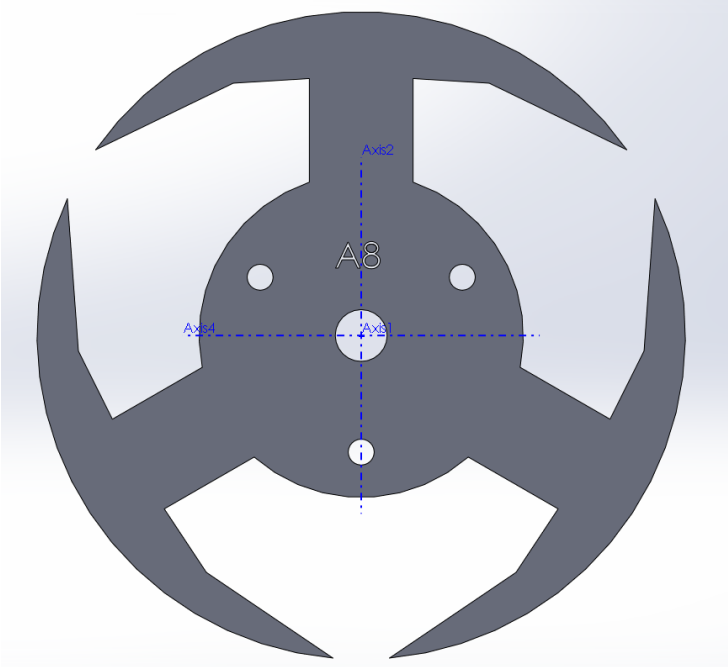
- Worked with 3 other people in designing and building a 5-bar robotic arm
- Designed and build an electric motor from water jet and 3d printed parts
- Robotic arm is controlled through a master/slave configuration
- Larger aluminum arms as the slave and a smaller scaled version 3d printed to be a master arm
- Used simple motors and attached sensors and current drivers in order to build custom servo motors

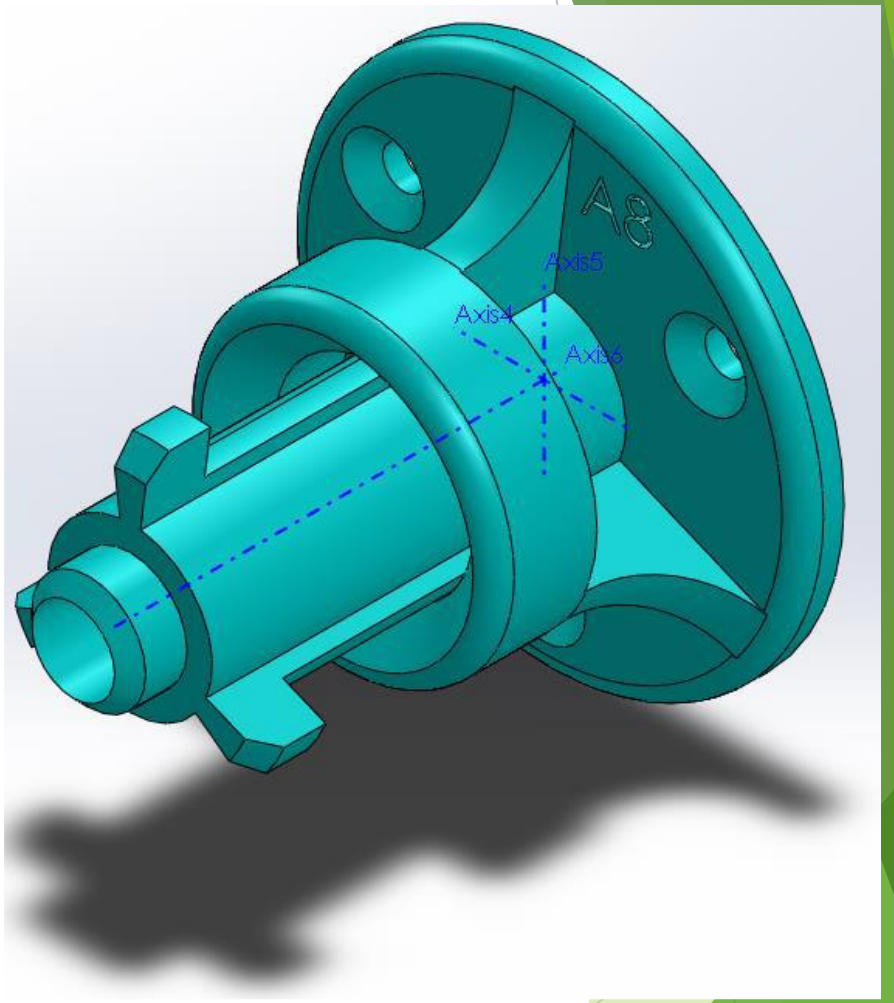




Electric Motor Design

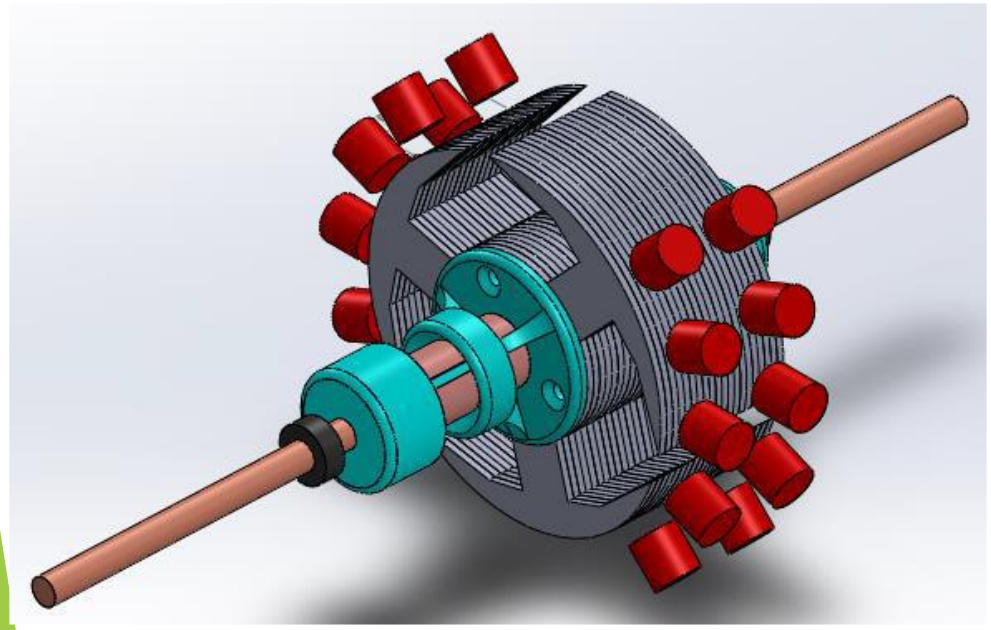
- Solidworks design of commutator and armature laminations
- Intricate parts were 3d printed and laminations were water jet cut

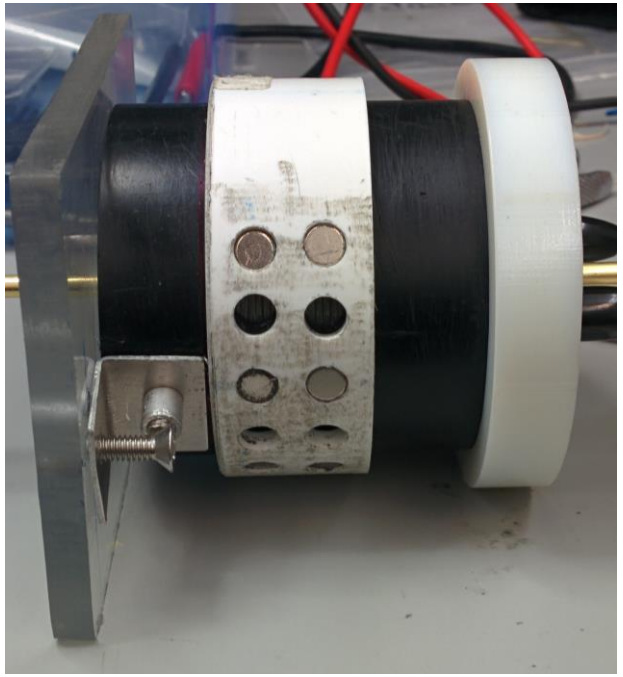


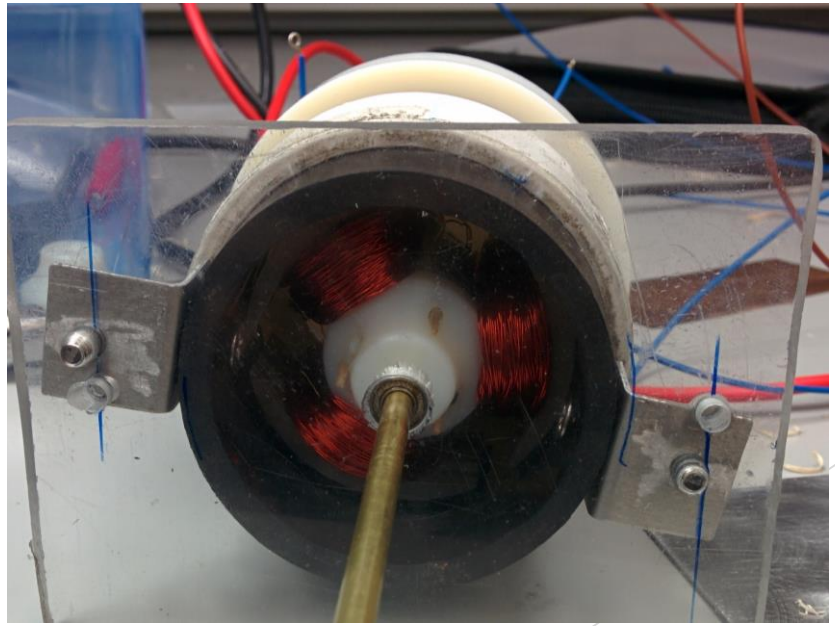


Electrical Motor

- Finished design and build of physical motor

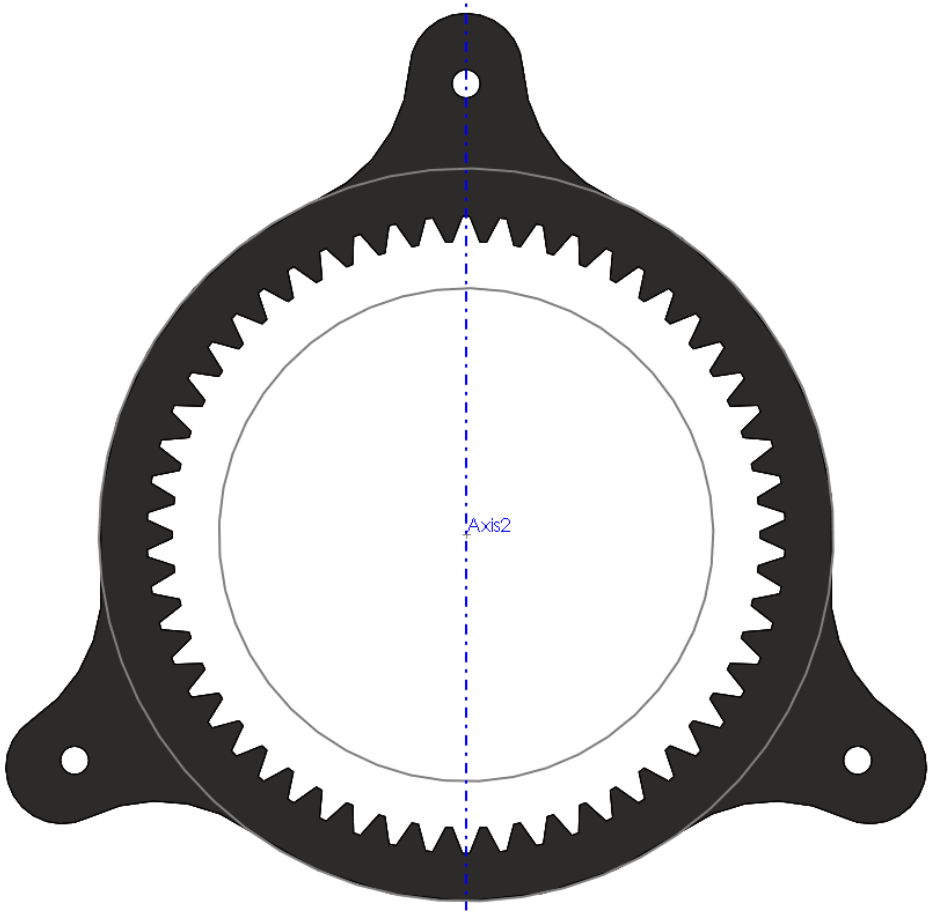


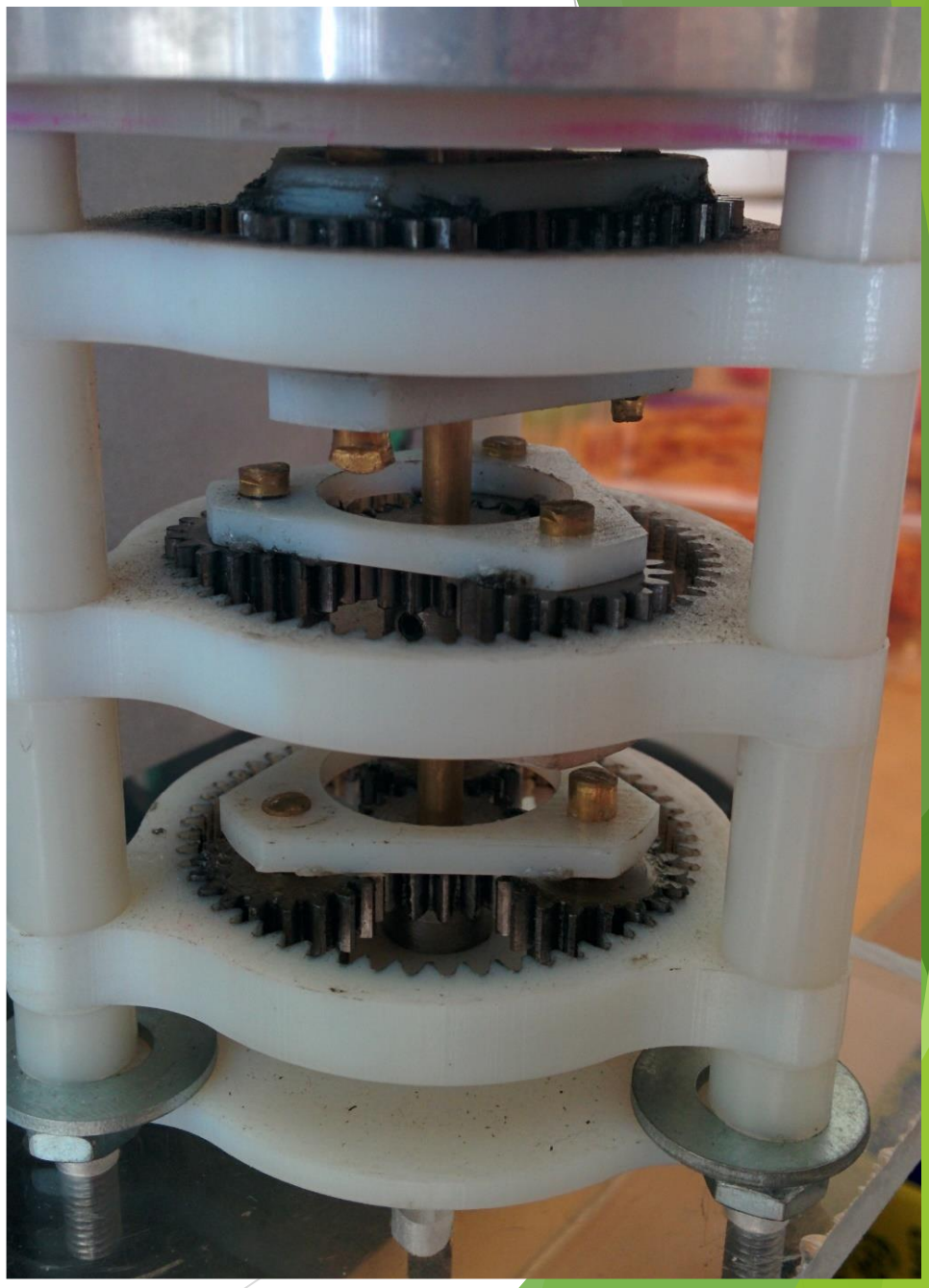




Gearing System

- A planetary gearing system designed in Solidworks and 3d printed





5-Bar Arm Robot

- Created a lifted platform and connected arms in a freely moving attachment
- Machined custom lock collars for the motors to attach to the arms