Virtual Quality Control Robot - Electrical Design

White-Paper **Updated - April 14, 2021**

ELEC 391 Team MotoGuzzi - Jeff Eom, Mark Gnocato, Yanyu (Grace) Zhang, ECE, University of BC

Abstract

A driver circuit designed to be placed on a printed circuit board is developed to control the speed and direction of four motors that attach to the arms and gripper of a quality control robot. Each motor requires its own driver circuit in order to operate independently. A driver circuit consists of a dual-input H-bridge that is used to control the speed and direction of the motor. This PCB is then mounted on the base of the robot and connects to the four motors, with one placed at each joint. An optical encoder is selected to measure rotation angle and this information is sent to the system's controller.

This report introduces the relevant nomenclature and describes an overview of the system electronics in Section 1. The motor selection process is reviewed in Section 2, and Section 3 describes the circuit design in detail. Section 4 presents the step response and model that will be used in subsequent steps of the design. Section 5 outlines the PCB configuration, followed by a look at the 3D model of the board in Section 6. Finally, Section 7 describes the encoder selection process, and Section 8 reviews the system RCGs. Refences and additional information are presented in Appendix A.

Nomenclature

SCARA	Selective Compliance Assembly Robot Arm
PCB	Printed Circuit Board
CAD	Computer-Aided Design
GBP	Gain Bandwidth Product
BJT	Bipolar Junction Transistor
NPN, PNP	Negative-Positive-Negative, Positive-Negative-Positive
TF	Transfer Function
CPT	Counts Per Turn
RCG	Requirements, Goals, Constraints

1. Electrical System Overview

Before designing the driver circuits, power source and microcontroller constraints are reviewed. An Arduino Leonardo is used which has a +5V rail. This means that an H-bridge is required for the motors to rotate in both directions. In order to keep the overall system frame small and inexpensive, one common power supply spanning from -18V to +18V is used. Thus, there is no need for multiple sources or stepping voltages up or down. A linear model is made to approximate the transfer function of the circuit for this virtual system. An optical encoder is used to measure position; this information is sent to the Arduino for appropriate PID response.

2. Motor Selection

The motor that was chosen is a balance of several parameters. Ideally the motor must be lightweight, small, minimize power consumption, and have a nominal torque that is able to support the remaining joints and gripper. As per Maxon Motor's Drive.Tech page [1], motors with precious metal brushes have a higher terminal capacitance, making them more responsive to sudden changes in voltage. Having a higher terminal capacitance also prevents increased no load currents and mitigates the motor temperature rise during operation. Fast response to change in voltage and reduced temperature are desirable parameters for this system.

The SCARA robot will be lifting small and lightweight objects and since the overall design is small, we are able to use the same motor at each joint and to control the gripper. The motor is an 18V Maxon DCX 22S Precious Metal Brush DC motor. The motor is 34mm long with a 22 mm diameter enclosure, and the shaft has a diameter of 3mm. The motor's max speed is 12400 rpm and has a stall torque of 14.5 mNm. The motor can be found on page 89 of the Maxon™ Motor catalog [2]. The motor is shown in Fig. 1

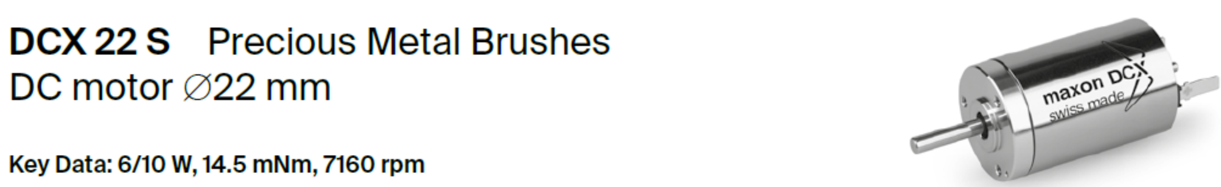


Figure 1: Maxon Motor

3. Circuit Design

The motor must operate in both the forward and reverse directions for a full range of motion with 3.5 degrees of freedom. An H-bridge circuit is used with two input ports that connect to the Arduino for directional control. The logic used to control the motor is shown in Table 1.

Pin 1	**Pin 2**	**Direction**
High	Low	Forward
Low	High	Reverse
Low	Low	Stopped
High	High	Stopped

Table 1: Directional Control from Arduino Pins

The motor is modelled using the terminal resistance found in the Maxon Motor Catalog, with a value of 8.75 Ω. Two op amps are connected to the input pins from the Arduino and connected to ±18 V to increase the voltage before it reaches the motor. Two THS4221D op amps were chosen due to their fast response and low distortion [3]. This op amp has a GBP of 120 MHz and a slew rate of 990 µV/s, which give it an extremely fast response and as a result a straightforward transfer function estimate.

Two PMBTA13 NPN Darlington Transistors [4] were used in combination with two PMBTA64 PNP Darlington Transistors [5]. These BJTs have a high voltage gain and deliver the necessary current for the motor, which has a stall current of 2.06 A. Feedback resistors are connected to the op-amp, and an optimal gain with minimum overshoot was found with R1 = 10 kΩ and R2 = 1 kΩ. Additional 10 kΩ pull-up and pull-down resistors are added to reduce the sensitivity of the circuit. The circuit is shown in Figure 2. The step voltage sources are used to model the input from the Arduino pins. Currents at each connection are measured using the probing tool. As an additional feature, bypass capacitors were added to limit noise in the feedback path and the overall response improved with less overshoot in the transients.

The H-Bridge is controlled by the Arduino, and some pseudo-code is shown in Figure 2A to describe the logic for controlling this circuit.

```
if desired angle > current angle    // motor operating in forward direction
    Pin1 = HIGH                     // output voltage regulated by PWM sets first pin HIGH
    Pin2 = LOW

if desired angle < current angle    // motor operating in reverse direction
    Pin1 = LOW
    Pin2 = HIGH                     // output voltage regulated by PWM sets second pin HIGH

else                                // motor is stopped
    Pin1 = LOW                      // output voltage regulated by PWM sets both pins LOW
    Pin2 = LOW
```

Figure 2A: Arduino Pseudo Code

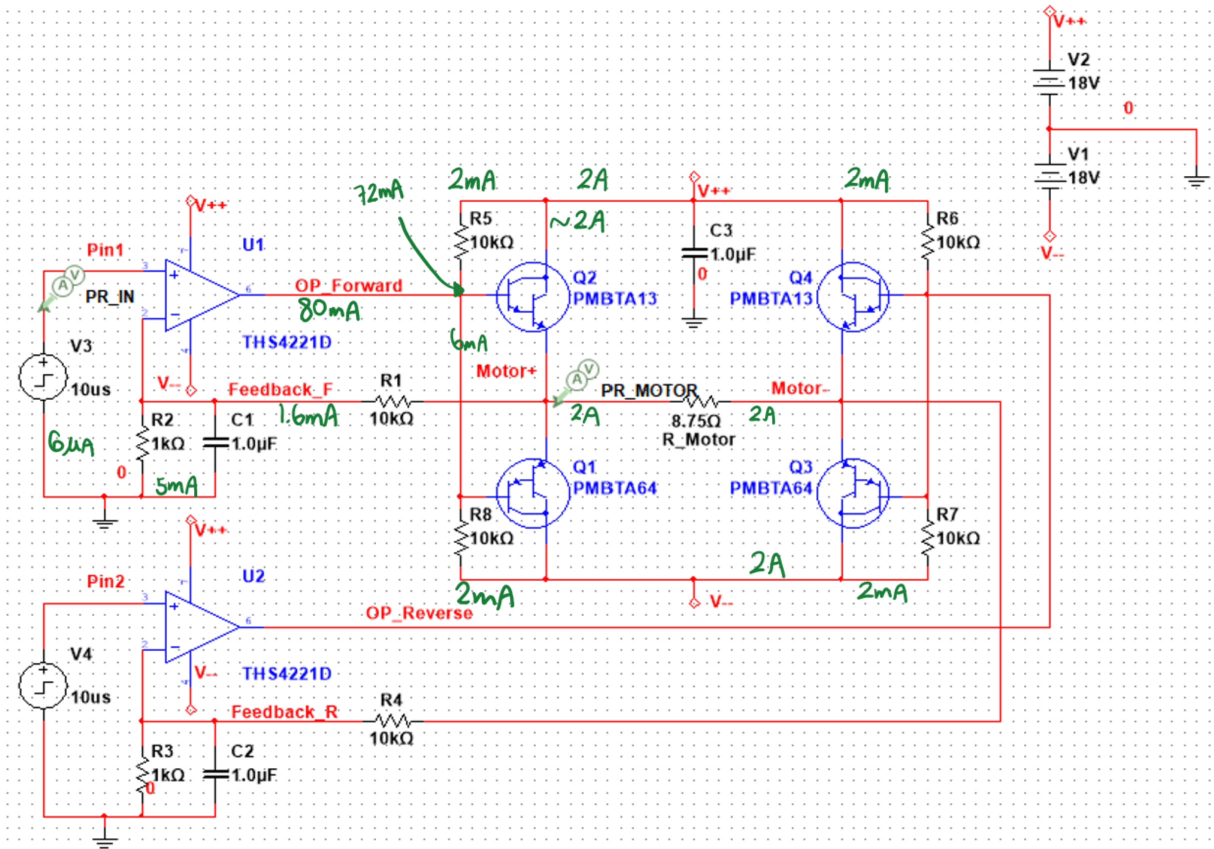


Figure 2: Driver Circuit

4. Driver Response and Linearization

The driver circuit in Figure 2 gives two different responses, depending on the inputs to the step sources, which correspond to the inputs in Table 1 above. The transient responses of the motor operating in the forward and reverse directions are shown in Figures 3 and 4A, respectively.

The reverse direction's voltage is taken as a differential voltage across the motor and as a result there is slight overshoot and settling before reaching a final value that is just above 16 V, which will suffice for the load and speed requirements when using the 18 V motor. The effect of the bypass capacitors can be observed by analyzing Figure 4B in the Appendix, where the response has a greater settling time.

Analysis of the transient plot indicates that the selected op amp gives a fast response with appropriate voltage gain for an 18 V motor. Figure 5 in Appendix A presents the near steady-state values of the motor's voltage and current, which are approaching the motor's stall torque and nominal voltage.

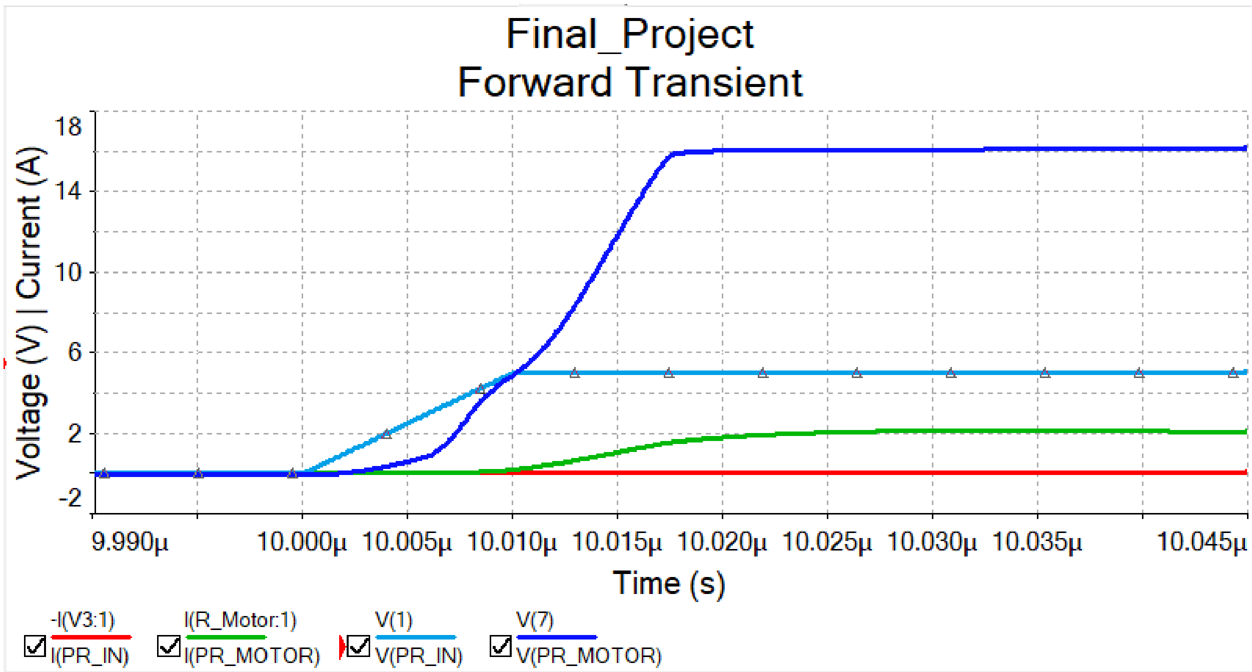


Figure 3: Driver Transient Response in Forward Direction

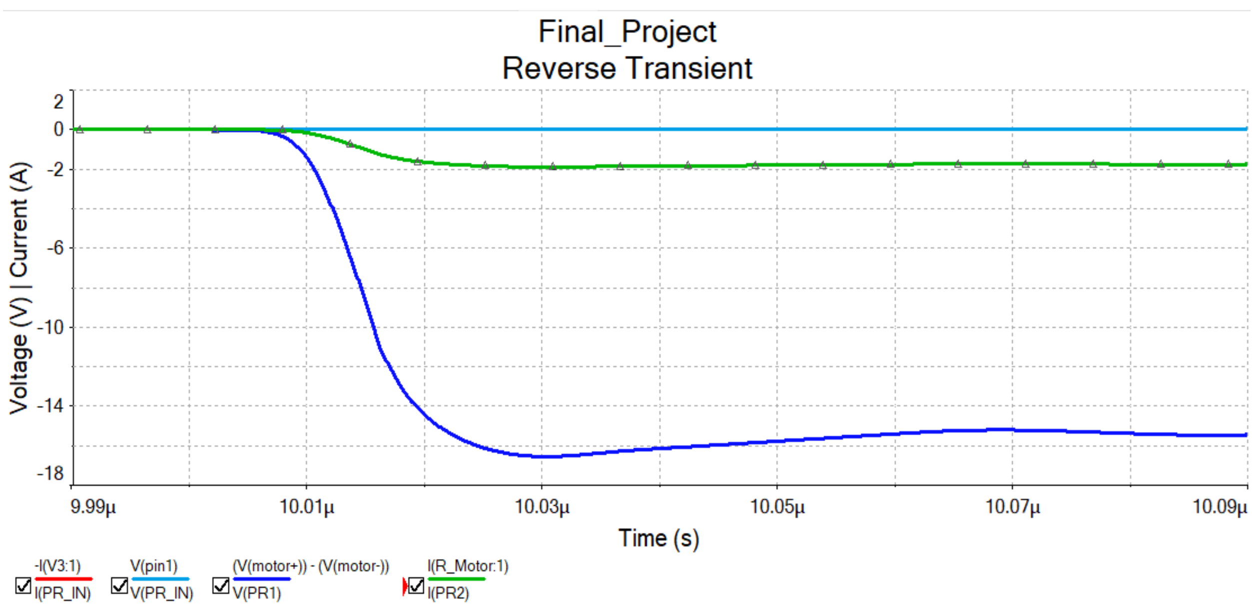


Figure 4: Driver Circuit Transient in Reverse Direction

Using the results obtained from the transient response of the driver circuit, MATLAB **tfest**() function may be used to estimate the appropriate transfer function. This method is preferred as it allows for accurate estimations and quick turnaround time when iterating designs. The estimated driver TF is then used in a model of DC motor as shown in Figure 6. The TF is shown in Figure 7 in Appendix A, with a fit above 90% using a second order estimation. A step response of the model using only a P controller is shown in Figure 8.

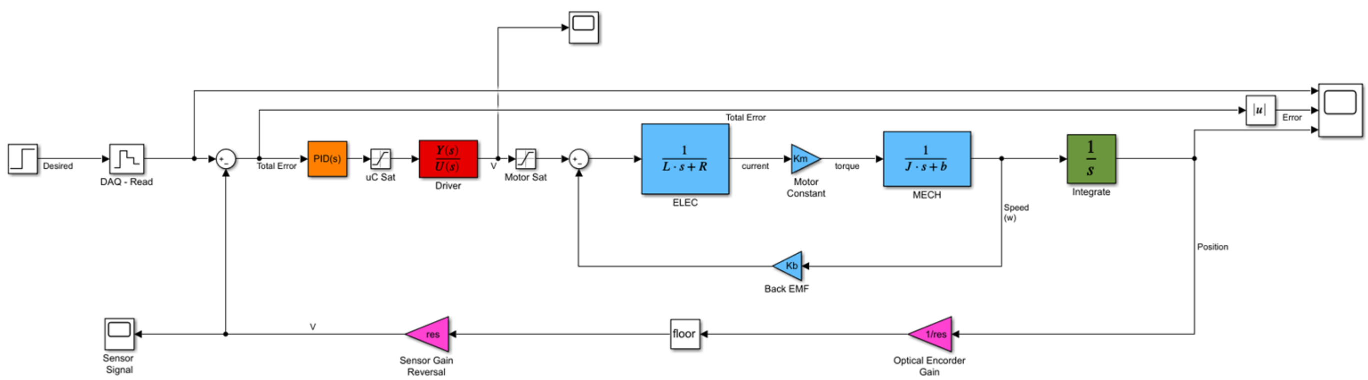


Figure 6: Motor Linear Simulink Model with Driver Circuit TF in Red

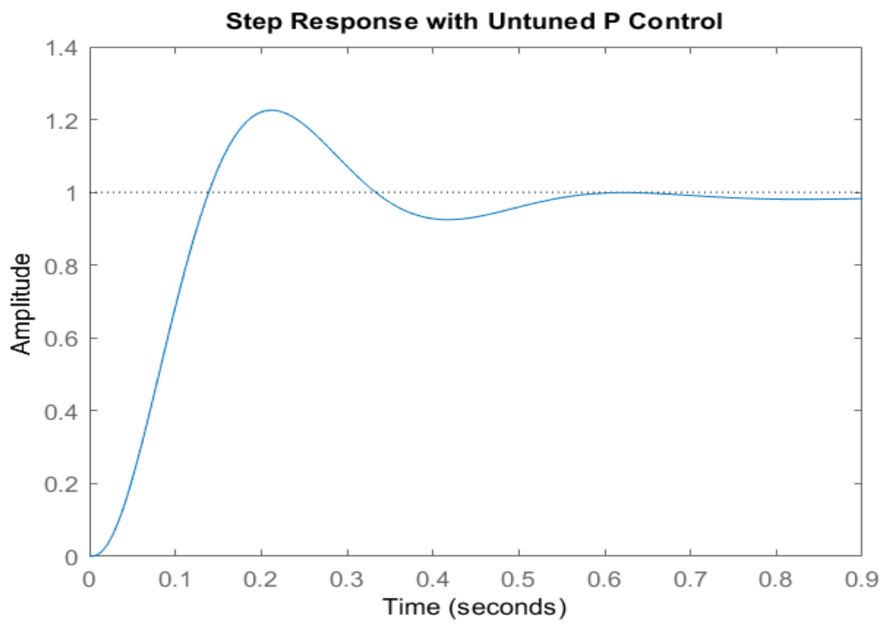


Figure 8: Linear Model Step Response with P Control Only

5. PCB Configuration

The PCB contains four H-bridge driver circuits, one for each motor. Since the circuits are identical, a similar approach was taken when laying out the four different drivers. An emphasis was placed on reducing board size and limiting the number of vias for ease of manufacturing.

As shown in the driver circuit in Figure 2, each node and input was probed and max current was recorded. The corresponding trace sizes can then be calculated for our PCB. Common manufacturing guides indicate that on the lower end, thickness of PCBs are 1-2 oz/ft^2 of copper. For 2oz of copper, spacing between traces of 8 mils is recommended [6]. When calculating trace width, the official DigiKey trace width calculator was used, which complies with IPC221 standards, and is sufficient for this application [7, 8]. An example of trace width calculation is shown in Figure 9 in Appendix A. The largest trace width was found to be 40 mils for 2 amps of current. These thick traces are connected between the power supply and the motors. All other traces carry current below 100 mA and therefore a trace width of 5 is appropriate and within manufacturing constraints when creating a PCB. The 2D drawing of the PCB is shown in Figure 10 and the 3D Ultiboard drawing is shown in Figure 11.

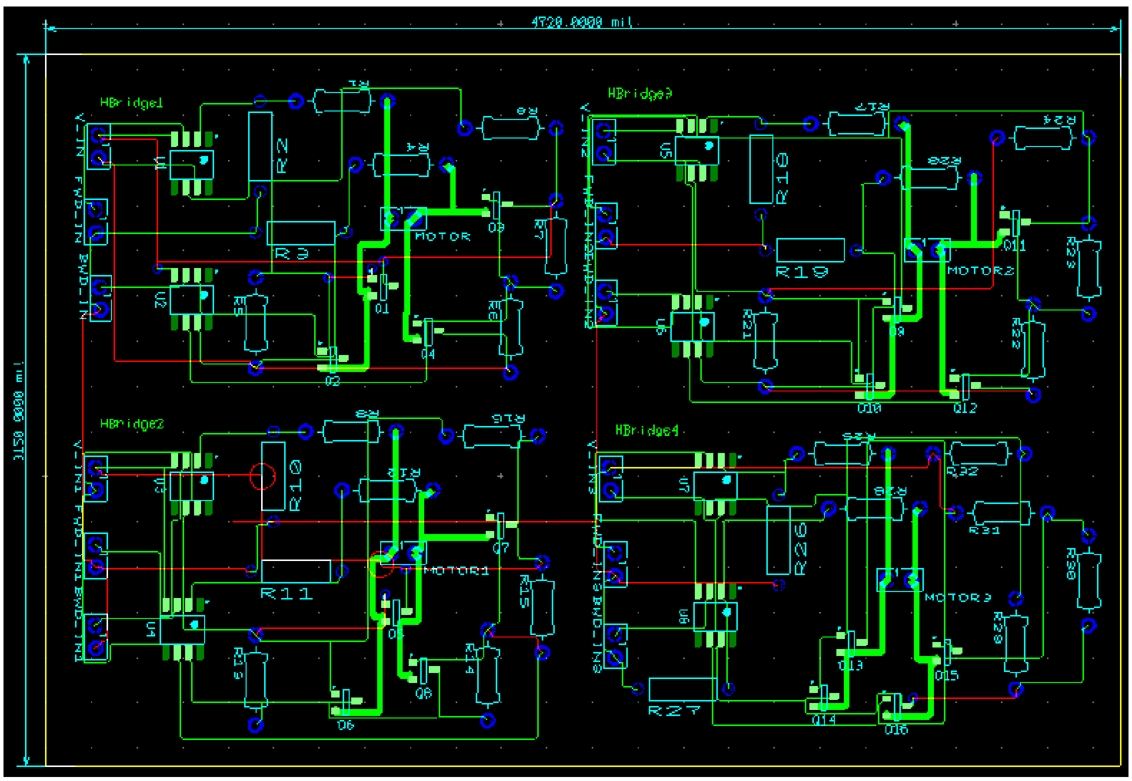


Figure 10: 2D Ultiboard Drawing of PCB

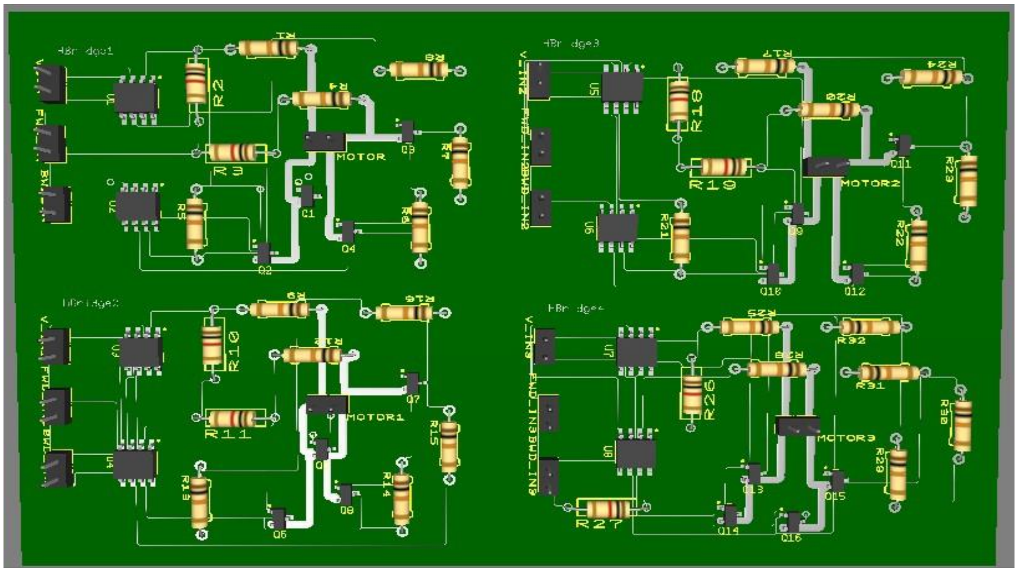


Figure 11: 3D Ultiboard Model of PCB

6. SolidWorks PCB Model

The PCB model is exported to SolidWorks as a 3D CAD model. The dimensions of the board are 120mm x 80mm. The SolidWorks model is shown in Figure 12A. The PCB may then be placed in a housing case shown in Figure 12B of Appendix A and then on the enclosure of the SCARA robot and connected to the Arduino output pins. The size of the 4 H-bridge PCB is comparable to the 2 H-bridge L298N as shown in Figure 13 in Appendix A. The L298N is a similar H-Bridge PCB that is commonly used for similar applications. This size comparison reinforces the small scale of the custom driver circuit.

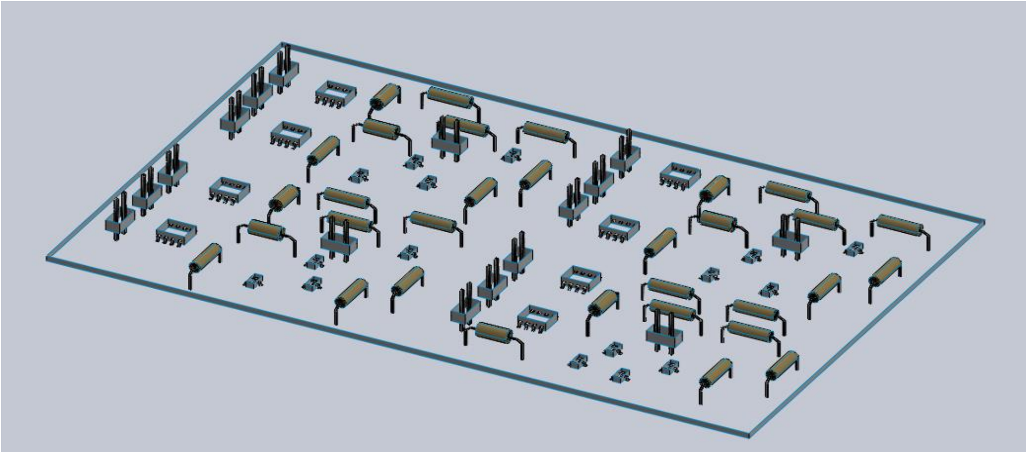


Figure 12A: SolidWorks shell model of PCB

7. Encoder Selection

Each motor position is measured by an optical encoder. The model chosen is a Maxon HEDS 5540 500 CPT, 3-Channel encoder. The part number is 110511 and has a 3mm shaft opening diameter. This motor was chosen because it is a quadrature encoder with relatively low resolution, giving us a good sampling rate of the position of each motor. Additionally, it is cheaper and easier to find than other alternatives.

The encoder dimensions are 30mm x 18.3mm. The CPT = 500 and is defined as being four times the number of pulses per revolution [9]. The CPT is used to find the resolution and number of windows in the encoder, as shown in Equations 1 and 2.

$$\text{Res} = \frac{360}{\text{CPT}} = \frac{360}{500} = 0.72 \qquad \text{Equation 1}$$

$$\#\,\text{Windows} = \frac{\text{CPR}}{4} = \frac{500}{4} = 125 \text{ windows} \qquad \text{Equation 2}$$

The sensor introduces noise into the system and since the measurements are discrete, the values that the controller reads are not exact, as indicated by the “floor” block in the Simulink Model in Figure 6. An encoder can then be mounted onto each of the motors to measure their individual positions for feedback into the Arduino. An image of the encoder is shown in Figure 14 and the part can be found on page 471 of the Maxon Motors Catalog.

Figure 14: Maxon Motors Encoder and CAD Model

8. RCGs

The electrical requirements include designing a custom driver circuit. This requirement is being met as the custom PCB contains 4 H-bridge driver circuits that are tailored directly to this application.

The constraints of the electrical portion include using a Maxon motor and Maxon encoder along with a single Arduino Leonardo. Additionally, the motor must operate at or below the nominal voltage and current. All constraints are being met, including the motor's operation below nominal values. The circuit provides the ability for the motor to draw up to the stall current, if necessary.

The main electrical goals include minimizing footprint, cost, and power requirements. 4 driver circuits have been placed on one PCB with the goal of minimizing the size of the board. The final size is comparable to that of a professional L298N dual H-bridge PCB; it is double the size with twice as many H-bridge drivers. The cost is minimized by using encoders that are among the cheaper options for the chosen motor, but still have a reasonable resolution for the application. Additionally, the PCB uses few vias and the overall footprint is small, which leads to ease of manufacturing and therefore reduced cost. Only one 18V power supply is required to power the four motors, so no additional electronics are required for stepping voltages up or down. The power requirements could be improved by using motors that require less voltage and are rated for higher torque.

References

[1] Maxon Drive.Tech, https://drive.tech/en/stream-content/precious-metal-vs-graphite-brushes

[2] Maxon Motor Catalog,
https://online.flippingbook.com/view/1042987/89/

[3] Op Amp Datasheet
https://www.ti.com/lit/ds/symlink/ths4221.pdf

[4] NPN Transistor Datasheet
https://assets.nexperia.com/documents/data-sheet/PMBTA13_PMBTA14.pdf

[5] PNP Transistor Datasheet
https://assets.nexperia.com/documents/data-sheet/PMBTA64.pdf

[6] Standard Copper Thickness for PCBs
https://pcbprime.com/pcb-tips/how-thick-is-1oz-copper/

[7] DigiKey Trace Width Calculator
https://www.digikey.ca/en/resources/conversion-calculators/conversion-calculator-pcb-trace-width

[8] IPC2221A Generic Standard on Printed Board Design
https://www.ipc.org/TOC/IPC-2221A.pdf

[9] CPR and PPR Definitions
https://www.cuidevices.com/blog/what-is-encoder-ppr-cpr-and-lpr

Appendix A

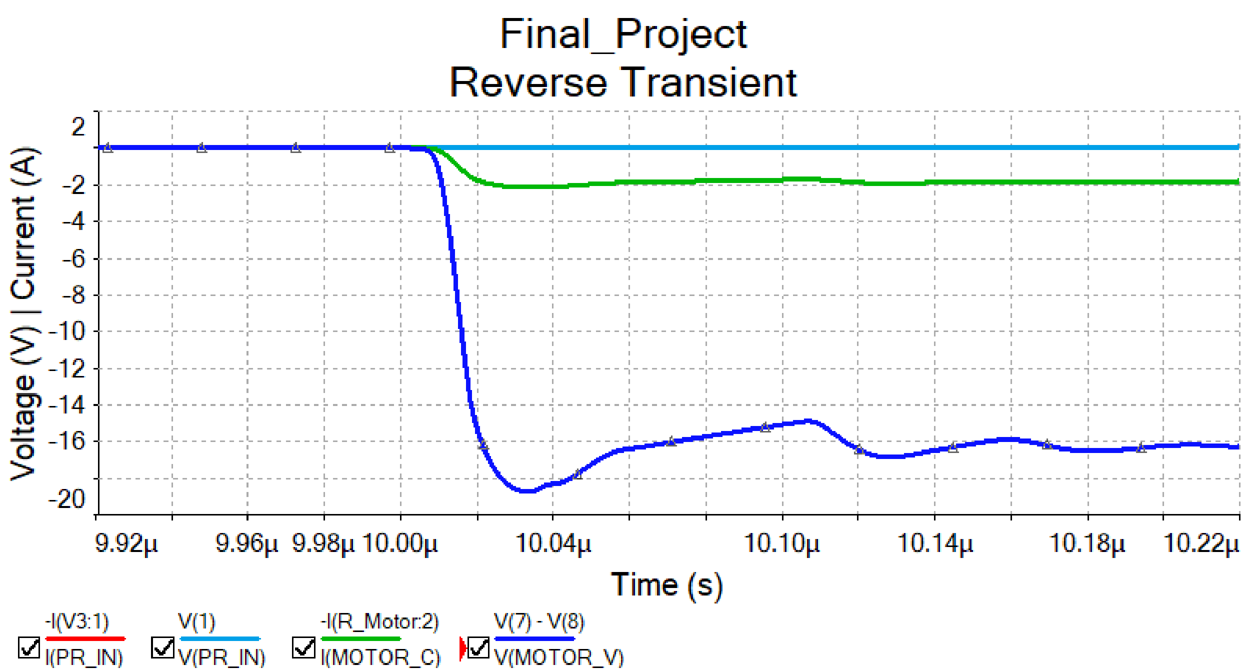
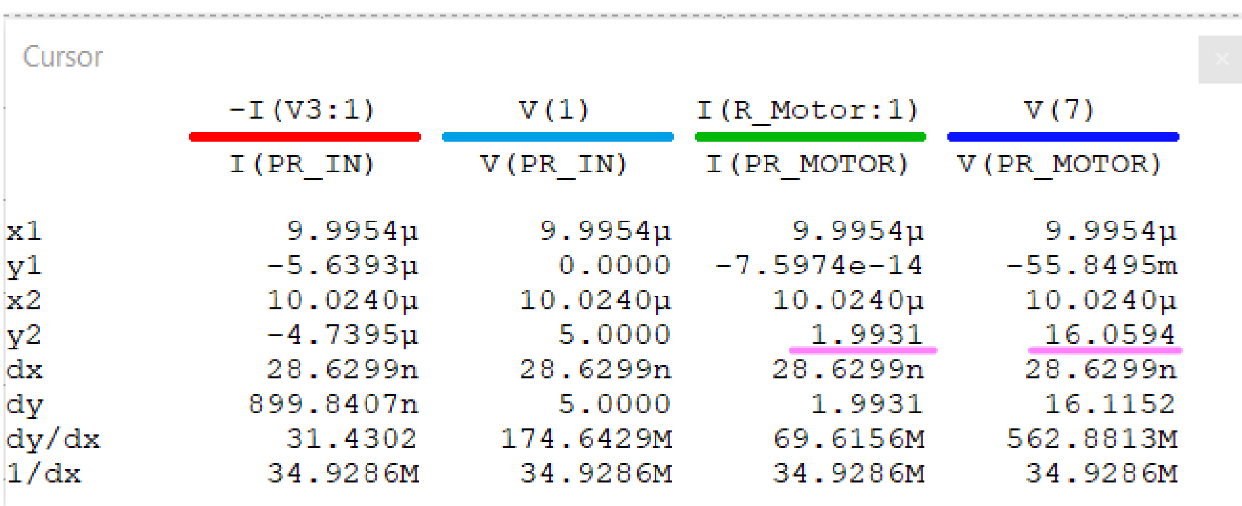


Figure 4B: Driver Circuit Transient in Reverse Direction without Capacitor



Cursor

	-I(V3:1) I(PR_IN)	V(1) V(PR_IN)	I(R_Motor:1) I(PR_MOTOR)	V(7) V(PR_MOTOR)
x1	9.9954µ	9.9954µ	9.9954µ	9.9954µ
y1	-5.6393µ	0.0000	-7.5974e-14	-55.8495m
x2	10.0240µ	10.0240µ	10.0240µ	10.0240µ
y2	-4.7395µ	5.0000	1.9931	16.0594
dx	28.6299n	28.6299n	28.6299n	28.6299n
dy	899.8407n	5.0000	1.9931	16.1152
dy/dx	31.4302	174.6429M	69.6156M	562.8813M
1/dx	34.9286M	34.9286M	34.9286M	34.9286M

Figure 5: Transient Near Steady-State Values

```
driver =

  From input "u1" to output "y1":
     72.47 s + 1.463
  ---------------------
  s^2 + 15.45 s + 4.745

Continuous-time identified transfer function.

Parameterization:
   Number of poles: 2   Number of zeros: 1
   Number of free coefficients: 4
   Use "tfdata", "getpvec", "getcov" for parameters and their uncertainties.

Status:
Estimated using TFEST on time domain data.
Fit to estimation data: 93.8%
FPE: 0.02393, MSE: 0.02391
```

Figure 7: Driver Transfer Function

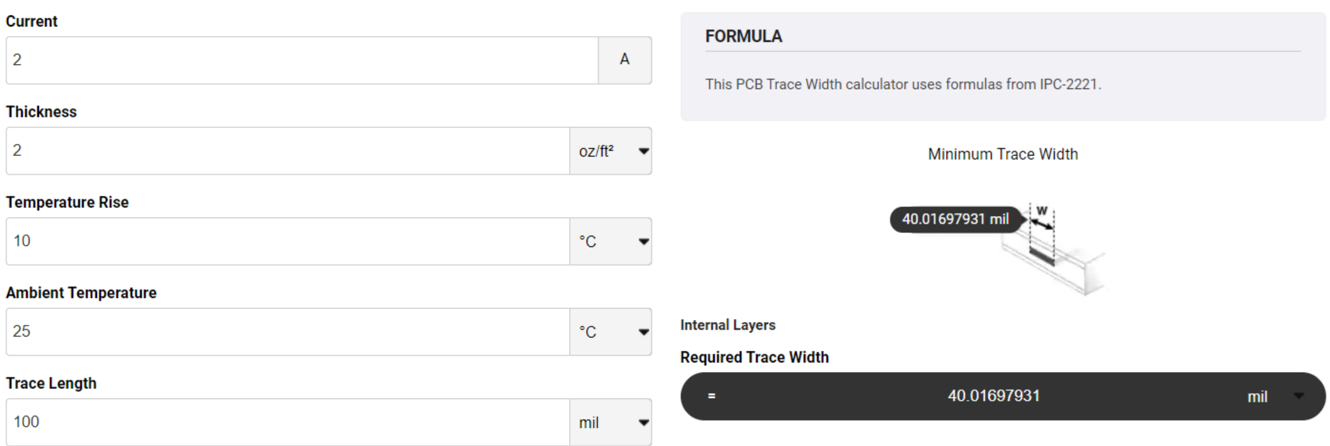


Figure 9: DigiKey Trace Width Calculator

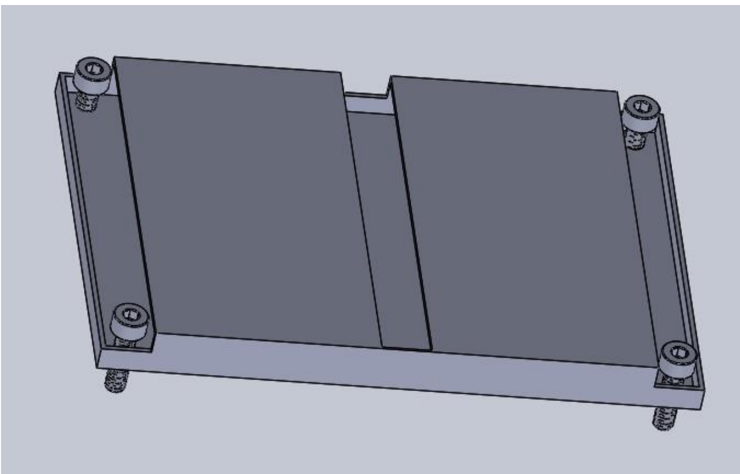


Figure 12B: PCB Housing

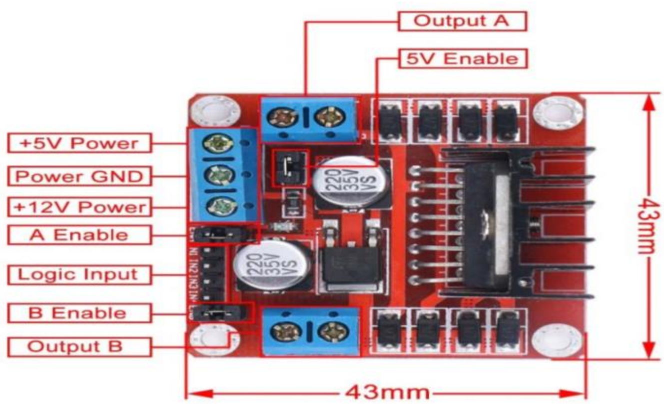


Figure 13: L298N Dual H-Bridge PCB