

# Electronics Report

## Fish Deboning Team A

### Software Used:

Simscape

Simulink

MATLAB

FritZing

## Table of Contents:

S.no.	Content	Page no
1	Simscape Electrical: Model and Results	3-10
2	Simulink: Control System for Speed control of Motor	11-12
3	MATLAB: Controller Verification	13-15
4	Fritzing & C++: IoT Circuit and Program	16
5	Python: Algo for Fish Head & Tail Detection	17
6	Human Machine Interface	18-19

## 1. Simscape Electrical : Model and Results

The entire electrical circuitry of the deboning machine is implemented in Simscape Electrical. The machine will work on single phase AC input of 230 Volts at 50 Hz Frequency.

Sub Models in the system:

- AC to DC Converter Circuit: A convertor circuit is implemented to convert the  $230(\sin 50t)$  input to constant 230V dc output to be given to the motors.

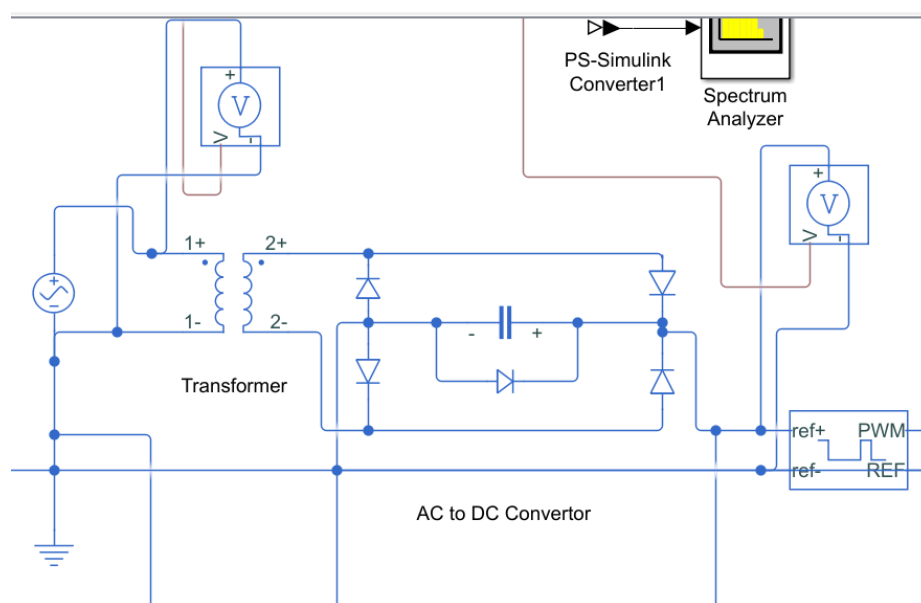


Figure 1: Converter Circuit

The bridge rectifier, in this case a full-wave rectifier. It is made out of individual discrete diodes 1N4007 with forward voltage of 1.1V. The idea is that we switch the negative AC pulses to positive pulses, and leave the already positive pulses there. There is some voltage loss due to the voltage requirements of the diodes, but it is minimal. The end result is a pulsed DC voltage, going from 0 to maximum voltage at 50Hz. We use a capacitor of 47uF across the '+' and '-' terminals to smooth out the ripples. As the voltage rises from 0 to max, the capacitor charges. When the voltage starts to drop, the capacitor discharges through the circuit but at a much slower rate, in effect holding the voltage up while the supply drops to 0 and then rises again. Once the voltage rises to where the capacitor voltage is, it recharges the capacitor and surges back to max again. As long as the ripple doesn't get below a certain

value. We can use that to power a voltage regulator, which simply stabilizes the wobbly input voltage to a specific output voltage. Full-wave rectifiers are better here than half-wave, since there is less time between the high and low pulses, resulting in a more stable output. Here we are getting a constant +230V dc output after the rectification.

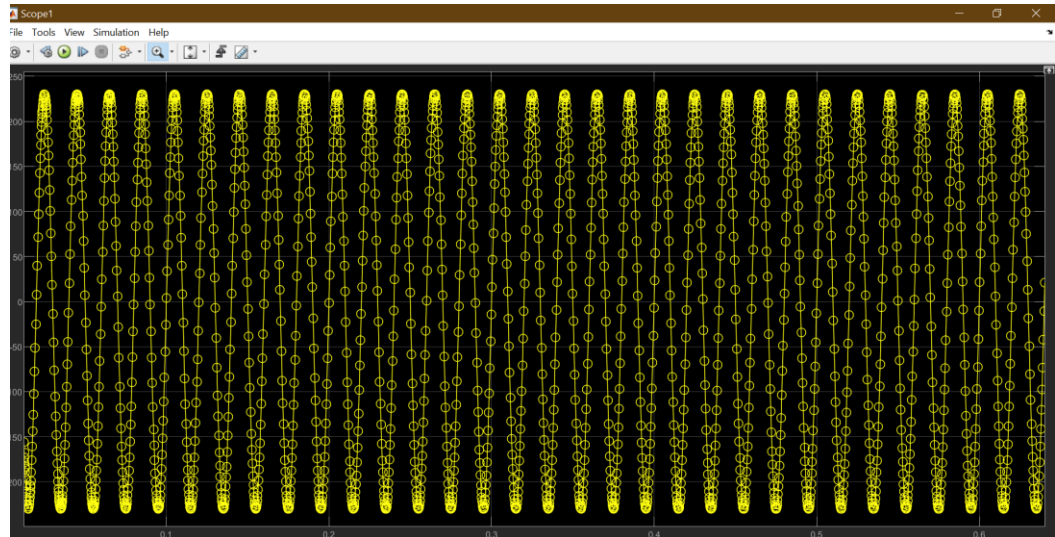


Figure 2: 230V, 50Hz AC Supply

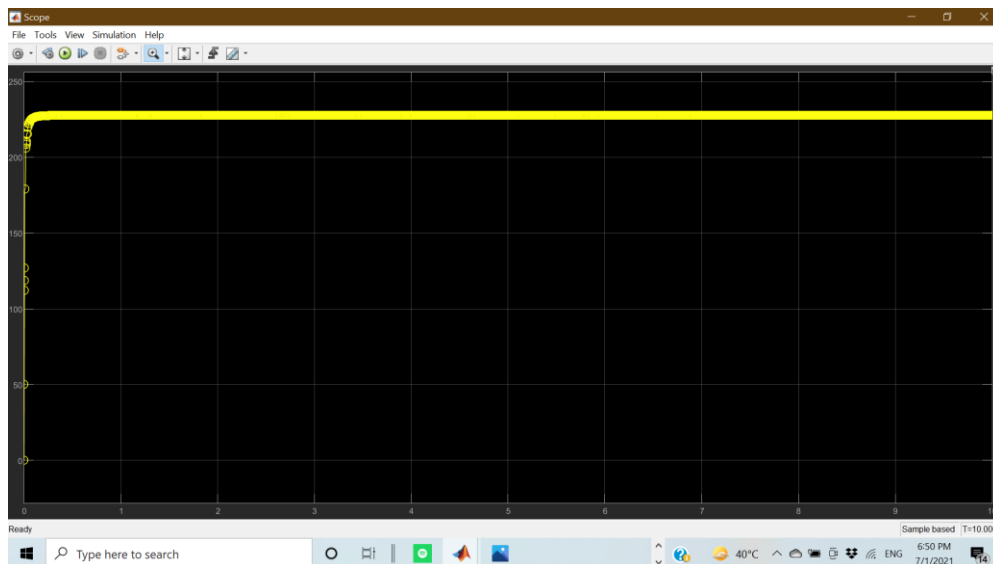


Figure 3: +230 V DC output

- **Controlled PWM Voltage and H Bridge:** The DC Voltage output is fed into a PWM controller with a switching frequency of 1000Hz of successive pulse. The output of the PWM controller is set at 5V which is to be fed to the H Bridge Industrial Motor controller rated at 5V DC input.

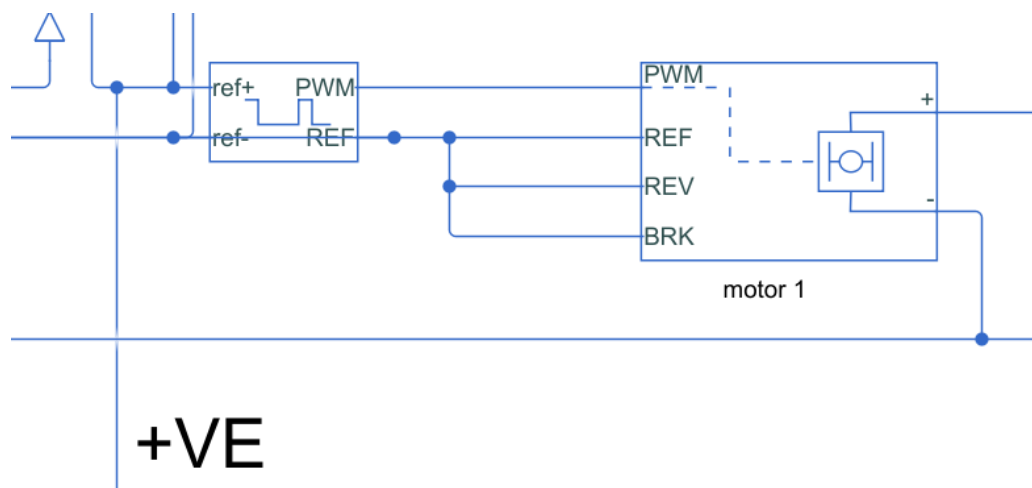


Figure 4: PWM and H Bridge

- **Industrial Motors:** For the purpose of Fin Removal, Descaling, Head& tail removal, Head collection, Filleting, Gutting, and to run Conveyor belts we have used industrial motors rated at 100W , 100 V , 5N-m Torque. The voltage output from the H bridge is amplified to 100 V to run these motors.
- **Fin Removal:** Two motors are used for Fin removal purposes. The current characteristics of these motors are plotted in the graphs. For Fin removal the motors use 1.0078 Amps of Current in its operation. Total power consumed =  $(2 \times 100 \times 1.0078) = 208 \text{ W}$

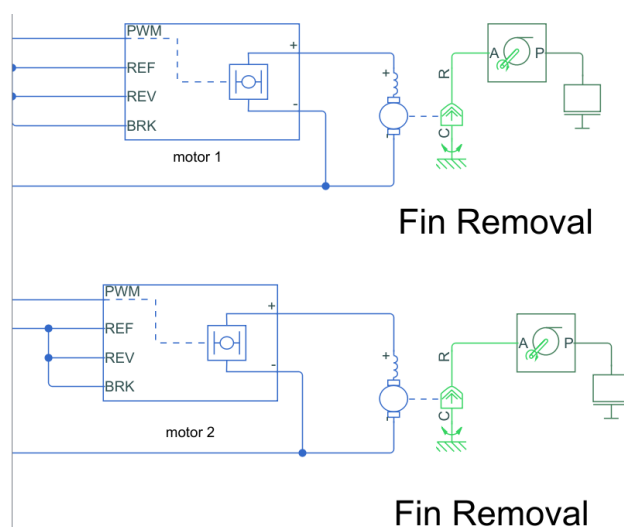


Figure 5: 2 Motors for Fin removal

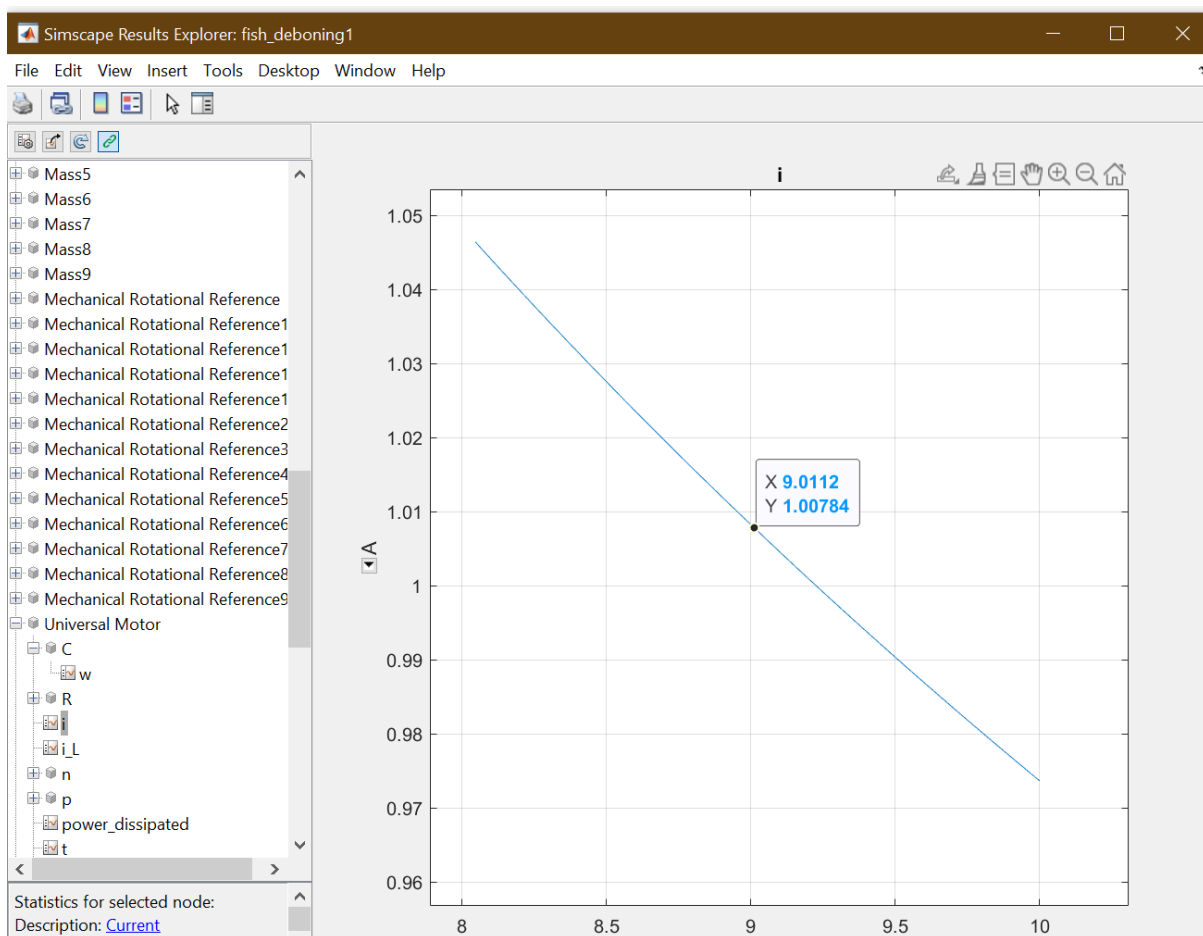


Figure 6: Current Characteristics of Motor

- **Descaling & Head, Tail Removal** : 1 motor each is used for these two purposes. The current characteristics of these motors are plotted in the graphs. For these motors, they use 1.0078 Amps of current in their operation. Total power consumed =  $(2 \times 100 \times 1.0078) = 208 \text{ W}$

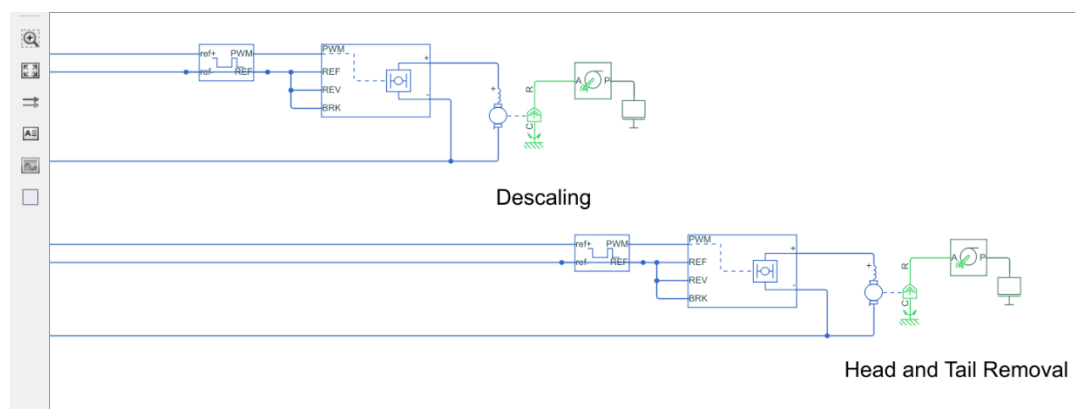


Figure 7: Descaling and Head, tail removal

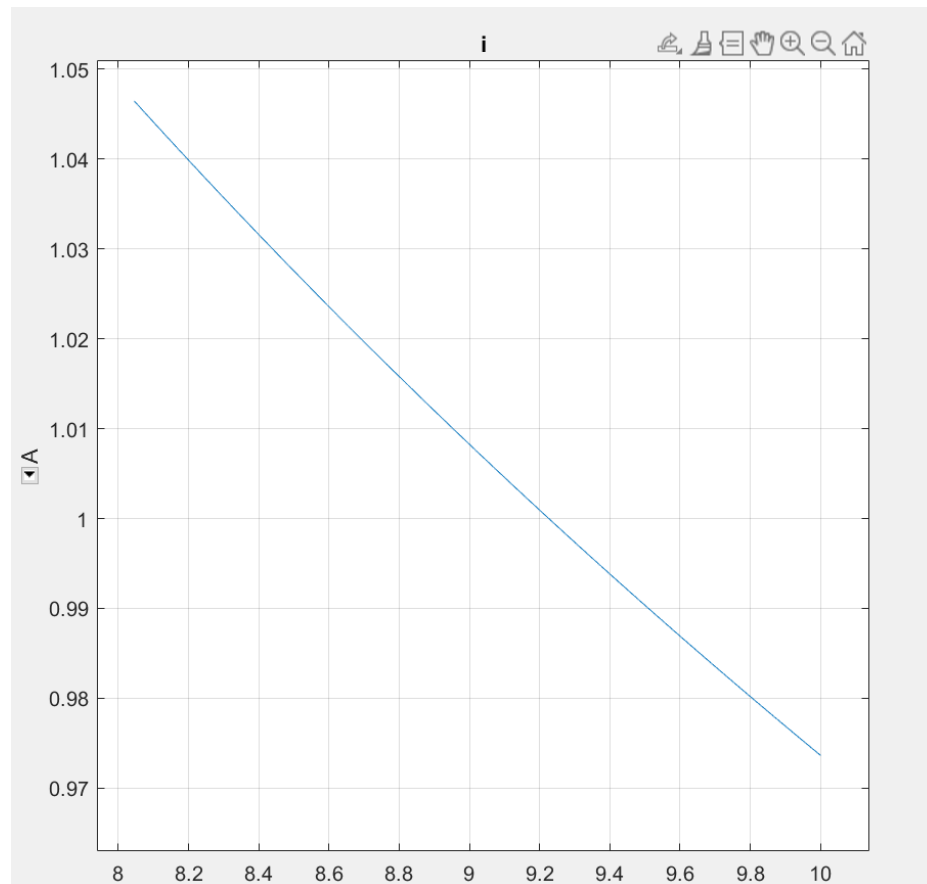


Figure 8: Current characteristics

- **Head Collection:** Fish head is also consumed; this motor acts as a flipper to collect the severed head of the fish and store them in a separate container. For this the motor use 1.04 Amps of Current in its operation. Total power consumed =  $(100 \times 1.04) = 104W$

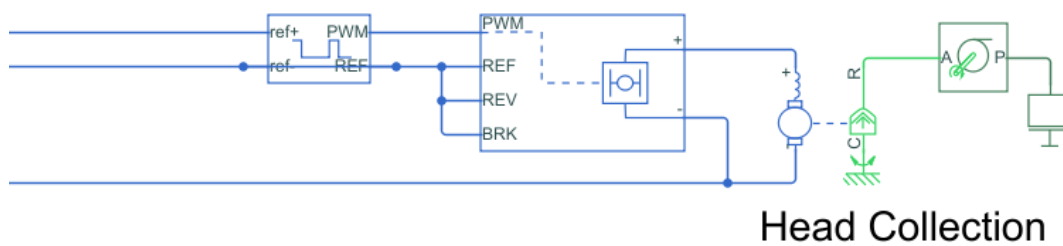


Figure 9: Head collection





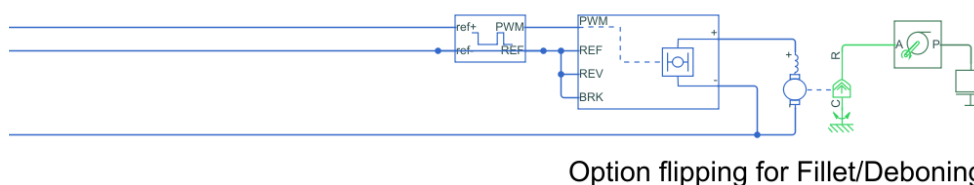


Figure 12: Fillet/Deboning

- **Deboning:** This motor runs the perforated drum for deboning this motor is rated at 230V. . For this the motor use 0.25 Amps of Current in its operation. Total power consumed =  $(230 \times 0.25) = 57.5W$

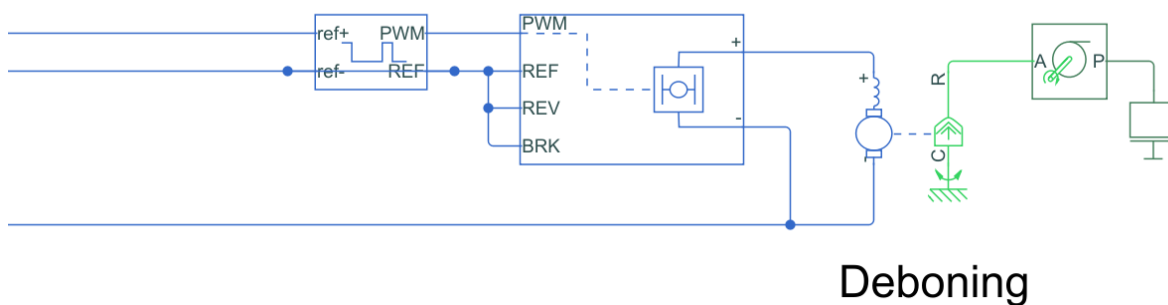


Figure 13: Deboning

- **Motors for running Conveyor belts:** In total 3 motors are required for running the conveyor belt at 60 Rpm. . For this the motor use 1.04 Amps of Current in its operation. Total power consumed =  $(3 \times 100 \times 1.04) = 312W$

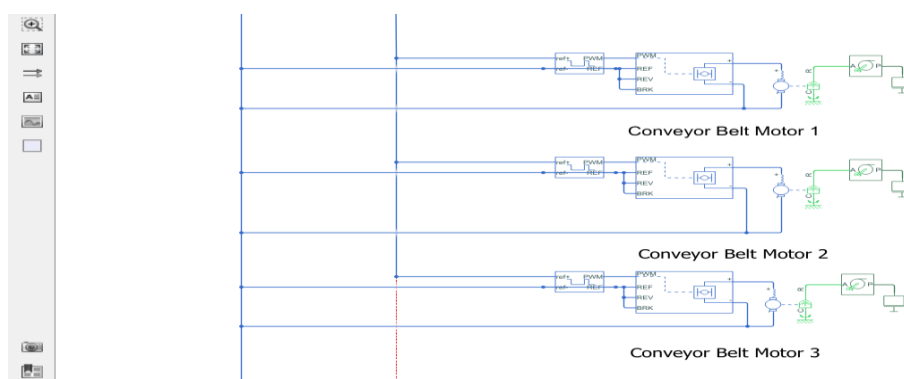


Figure 14: Conveyor motors

Total power consumed by all motors : 1293 Watts

Power rating of Machine : 1293-Watt hour

## 2. Simulink: Control System for Speed control of Motor

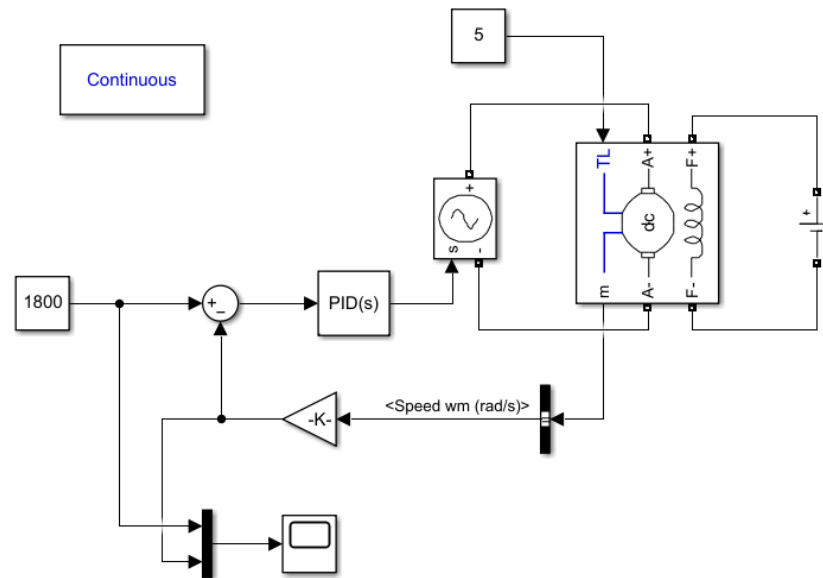


Figure 15: Control System

A closed loop PID based control system is designed for the optimal speed control of the Motor. The constant is set at 1800 Rpm which is required for the deboning process. Here the Dc machine is fed with variable voltage supply its RPM output is obtained and is multiplied by specific gain constant to convert it to RPM. This acts as a feedback signal to the control system this is subtracted from the constant RPM we want i.e. 1800 RPM which produces the error signal , this error signal is fed to PID block where the gains  $K_p$  ,  $K_d$  ,  $K_i$  are tuned accordingly to reduce the settling time ( $T_s$ ) and minimize the maximum overshoot ( $M_p$ ). The actuating signal obtained produced is given to the variable voltage supply. The results are plotted using a scope.

$$K_p = 0.851320759653448$$

$$K_i = 11.6736558254255$$

$$K_d = 0.014231201926504$$

The stable control system parameters obtained:

$$\text{Rise Time} = 0.0615 \text{ secs}$$

$$\text{Settling Time} = 0.201 \text{ secs}$$

$$\text{Overshoot} = 6.8\%$$

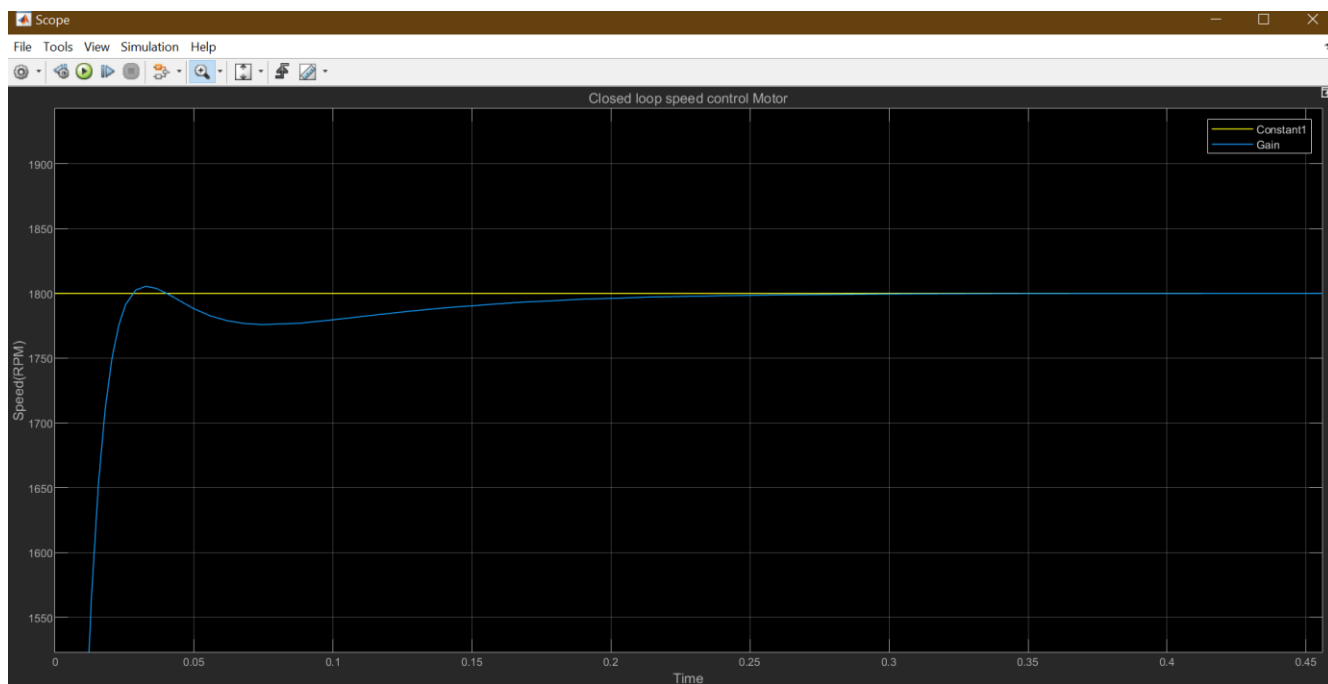


Figure 16:Control system response

### 3. MATLAB: Controller Verification

A live script is written in MATLAB with certain motor constants , through this we calculated the continuous time transfer function of the motor

```
J = 0.01;
b = 0.1;
K = 0.01;
R = 1;
L = 0.5;
s = tf('s');
P_motor = K/((J*s+b)*(L*s+R)+K^2)
```

```
P_motor =

          0.01
-----
0.005 s^2 + 0.06 s + 0.1001
```

Continuous-time transfer function.

Now a PID Controller is added to the system and its output was obtained

```
Kp = 100;
Ki = 200;
Kd = 10;
C = pid(Kp,Ki,Kd);
sys_cl = feedback(C*P_motor,1);
step(sys_cl, 0:0.01:4)
grid
title('PID Control')
```

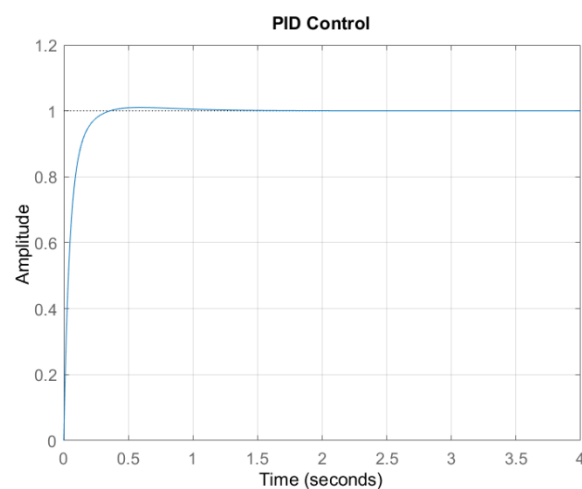


Figure 17:PID plot

Root Locus of the open loop transfer function is obtained for learning about absolute stability of the system.

```
%Root Locus%
```

```
controlSystemDesigner('rlocus', P_motor)
```

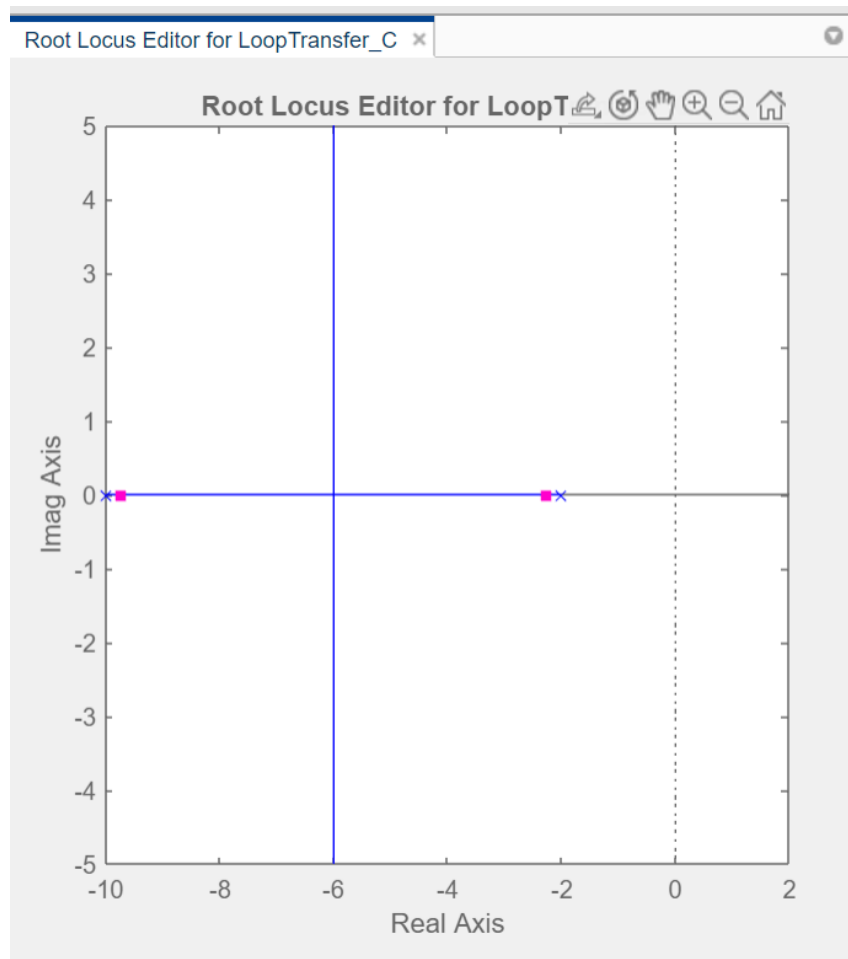


Figure 18:Root Locus

Since the Root locus is towards the left of the imaginary axis hence the system is stable.

Bode Plot is also obtained for the system for information regarding relative stability of the system. The gain margin and phase margins are also calculated

```
bode(P_motor)
grid
title('Bode Plot of the Original Plant')
```

```
margin(P_motor)
[Gm,Pm,Wcg,Wcp] = margin(P_motor)
```

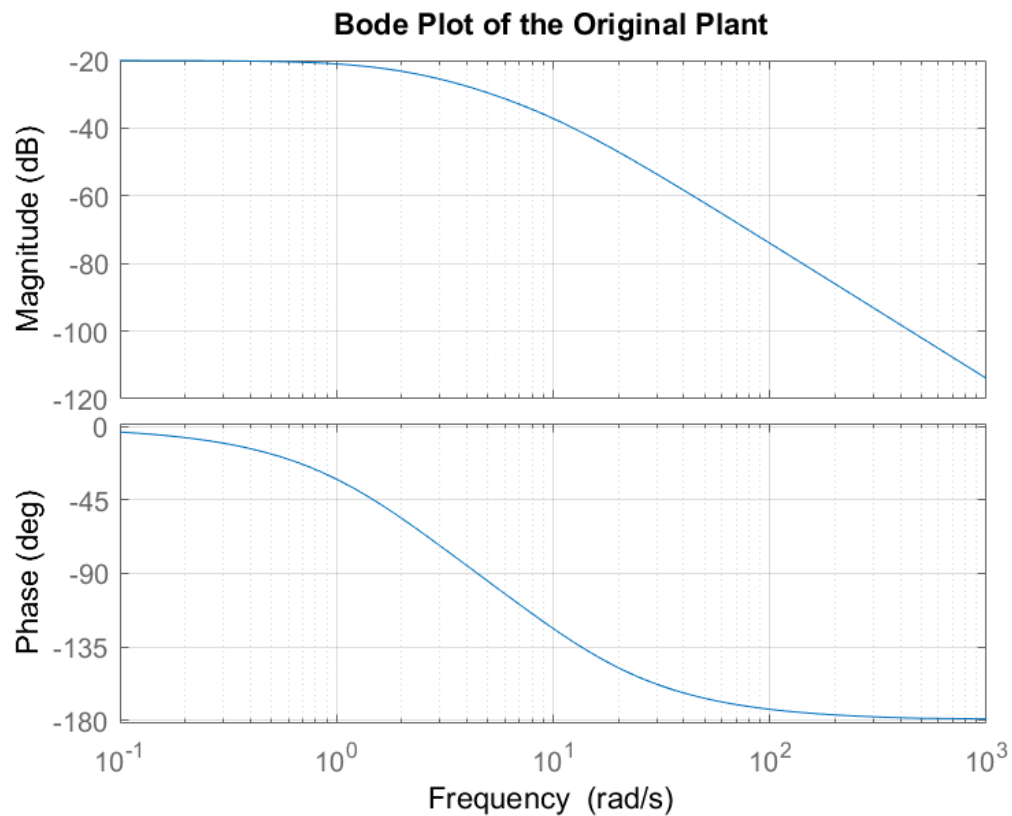


Figure 19: Bode Plot

Gain and Phase margins:

```
Gm = Inf
Pm = Inf
Wcg = Inf
Wcp = NaN
```

Since there are coming out to be Infinity so our motor will always be stable throughout its functioning in the machine.

## 4. Fritzing & C++: IoT Circuit and Program

The amount of fish weight obtained is sensed thorough a load cell controlled by an ESP8266 WIFI Module, this module will send the real time weight data to MATLAB ThingSpeak an online IoT analytics platform, The farmers can know about the amount of fish meat produced and compare their produce. It will also have a temperature sensor LM35 to keep track of the temp of fish obtained.

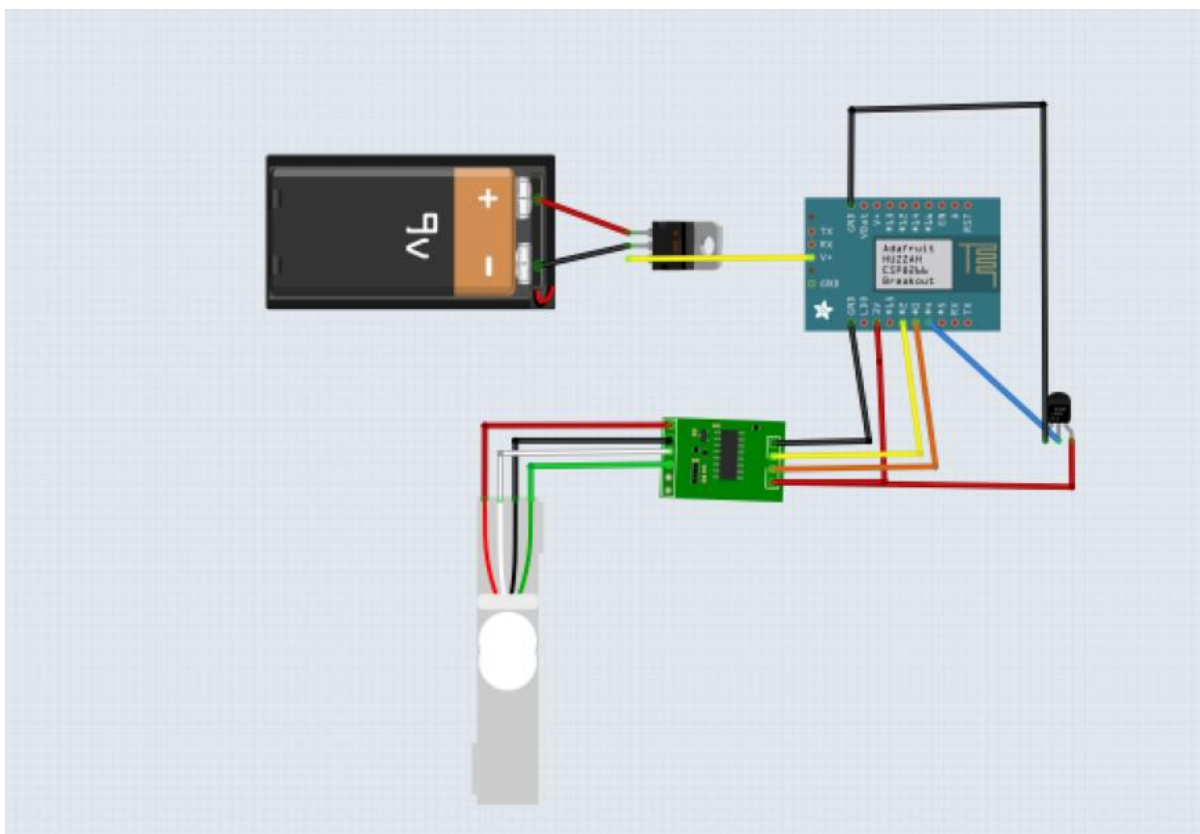


Figure 20: IoT Circuit

The program for sending the data to ThingSpeak platform:

GitHub Profile(Private Repo): <https://github.com/shivu926/FishDeboning>



## 5. Python: Algo for Fish Head & Tail Detection

The program for detecting head and tail of fish uses IR Sensors controlled through a Raspberry Pi Module, the delay for switching on/off the motor for beheading is calculated using a time counter.

Code available at: <https://github.com/shivu926/FishDeboning>

## 6. Human Machine Interface

The HMI is placed at the starting of the machine for the operator to decide whether the desired output is fillet or deboned mush.



Figure 21: HMI

Depending upon the choice the output will be produced.

Connected of STM32 Controller

```
int x;
printf("%s", "Enter the choice 1. Fillet 2.Mush")
scanf("%d",x);
switch(x):
{
case 1:
    GPIO.output(Motor9.HIGH);
    break;

case 2:

    GPIO.output(Motor9.LOW);
    break;

Default:
```

```
GPIO.output(Motor9.LOW);  
break;  
}
```