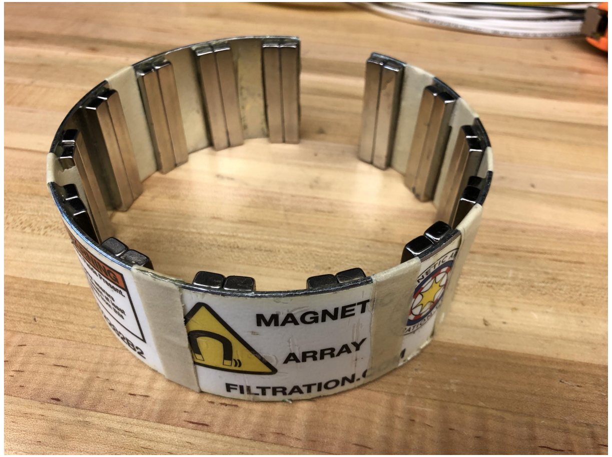
Gauss sensor for magnet array filter

Team : sdmay20-27
Members: Vishal Patel, M. Lutfi Latip, M. Aiman Zulkefli, Irfan Rafie, Weinee Long
Advisor : Dr. Mani Mina
Client : Magnet Array Company, Dennis O'neel

Problem statement



- contaminants that can build within a mechanical system
- filter in a vehicle is ineffective for small particles
- Magnetic array filter used to help trap smaller contaminants in the filter
- No efficient method of knowing when to change the filter.

GOAL :

- Design a sensor that could detect the change in magnetic
- Alert the user time to change filter.

Lutfi

High level overview of the project

1. **Interviewing**: gain information from the client

2. **Problem Statement:** Defining all the problems stated and discussed that helps in guiding the direction we are going for in getting the expected outcome for the project.

3. **Research**: Research and data collection conducted that are necessary in solving the problem statement defined in previous step

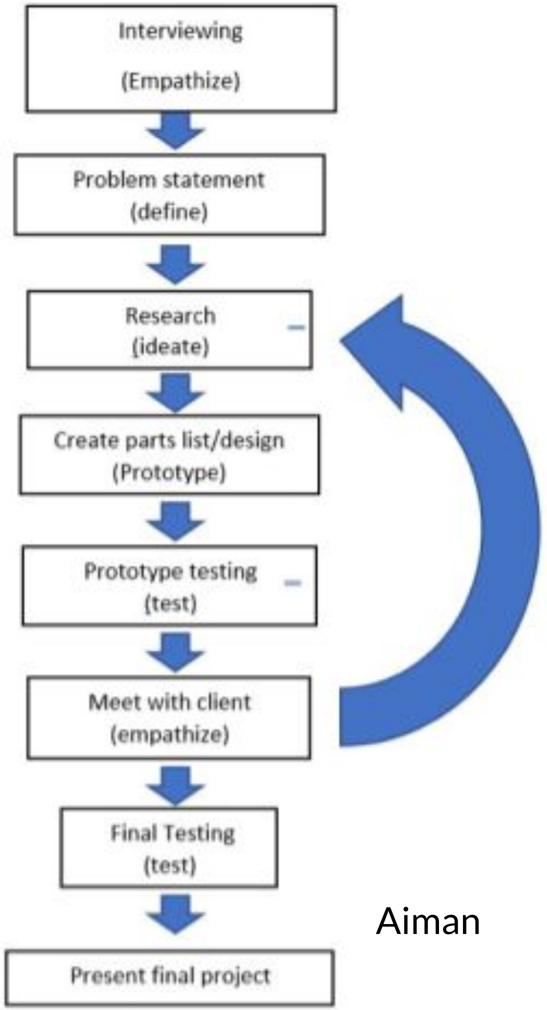
4. **Prototype Design:** The team will initiate an initial solution and possible prototype design based on the ideas generated earlier

5. **Prototype Testing:** The team conducts rigorous testing on the prototype, analyze the result and provide deduction based on the result whether the solution works

6. **Meet with Client:** The team will present the outcome of the testing to the client and discuss whether the conditions are met. If it is not, the team will go back to Research step

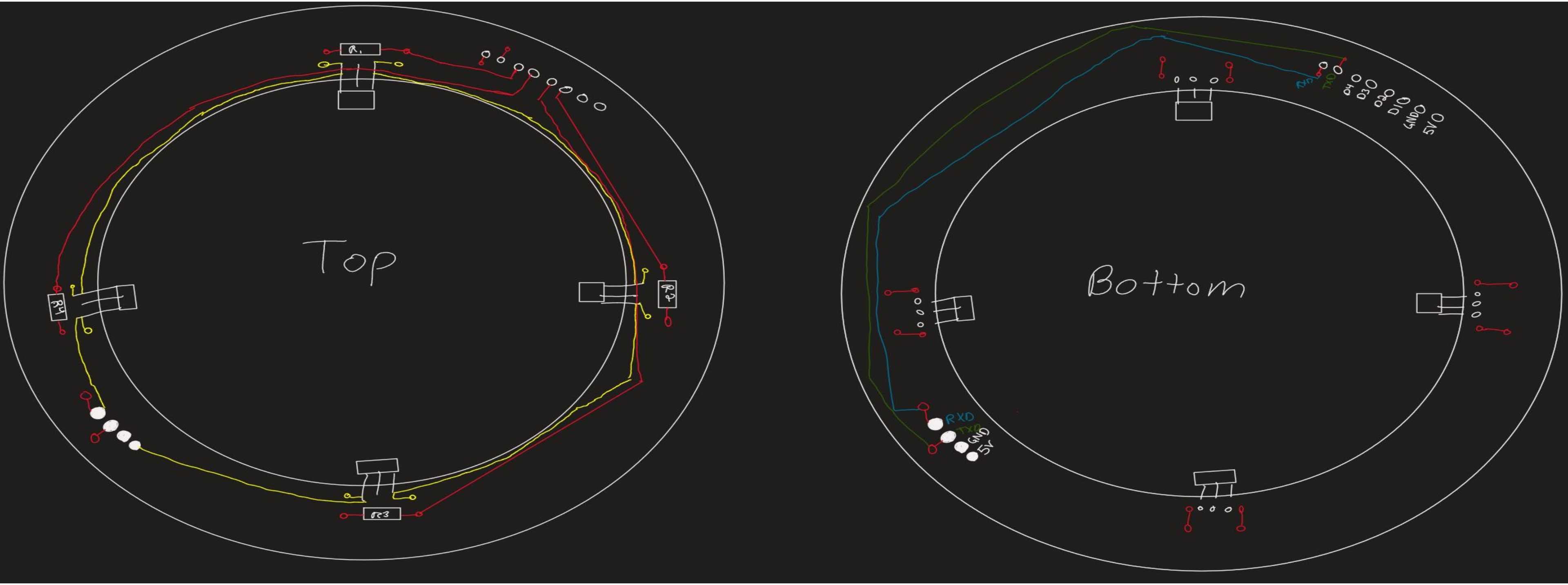
7. **Final Testing:** If the client is satisfied with the outcome, the team should conduct a final testing on the final product to ensure the quality of the product before finalizing the result

8. **Present final project**



Aiman

Conceptual Sketch



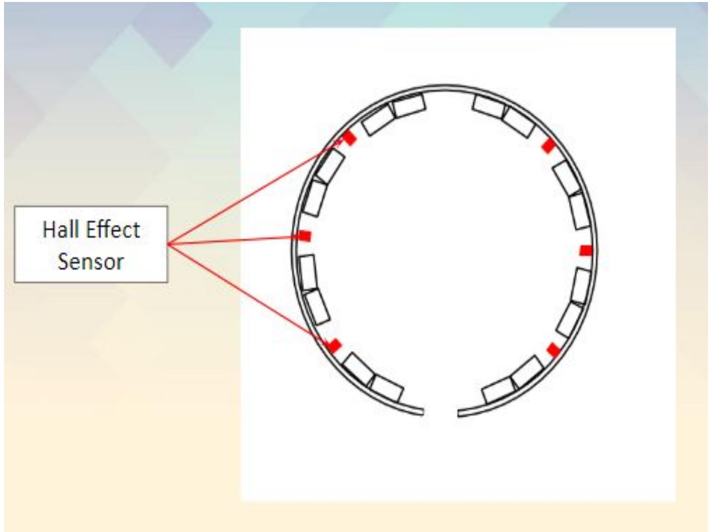
Aiman

Functional Requirements

Our product must:

- Detect magnetic particles, iron (sensitivity **0.05 µm**)
- Detect when then buildups reach the threshold
- Indicate the time to change the filter
- Indicate when there is a significant change in the buildups of particle

- **Additional functionality:** detect brass and aluminium particles



Wei-Nee

Technical/Other Constraints/Considerations

<u>Non-functional requirements:</u>

- Reduce time and money on gradually checking, changing, and replacing filter.
- Make sure iron particle is not going through system
- Increasing engine life
- Improve efficiency of overall maintenance process

<u>Constraints:</u>

- Not able to test the field with increasing temperatures accurately
- Does not have direct access to a vehicle

Wei-Nee

Market/Literature Survey

- Currently on the market, there is no way of knowing when the filter has reached a certain threshold to where particles will not be caught.

- This is because as more of the wear debris metal particles are trapped by the magnetic array filter, the particle build ups causes a deterioration in the ability to attract any more of the wear particles inside the oil system.

Wei-Nee

Potential Risks & Mitigation

Accuracy

- Largest risk is accuracy of the sensors. The environment has large fluctuation in temperature and also repetitive mechanical vibration.

Equipment

- Risk in using hydraulic test bench. It is used extensively for data collection. Despite training, there is still high risk in using it. (Eg: oil spills, high temperature oil leaks)

Knowledge of Area

- Knowledge on Hall Effect sensors. We are taught how they work but do not have in depth hands-on experience with them. This can cause some delays in reaching milestones for the project.

Wei-Nee

Resource Requirements for Prototype Unit

Part Name	Quantity	Price per unit	Total Price
DRV5053	12	$0.88	$10.56
DRV5056	12	$1.33	$15.96
DRV5057	12	$1.78	$21.36
CD74HCT4067-Q1	3	$0.61	$1.83
Capacitors	6	$0.34	$2.04
Resistors	36	$0.1	$3.6
Collar Fitting(3D Printed)	1	$0.65 with ABS	$0.65
Arduino Uno	1	$20.50	$20.50
		Total	$82.74

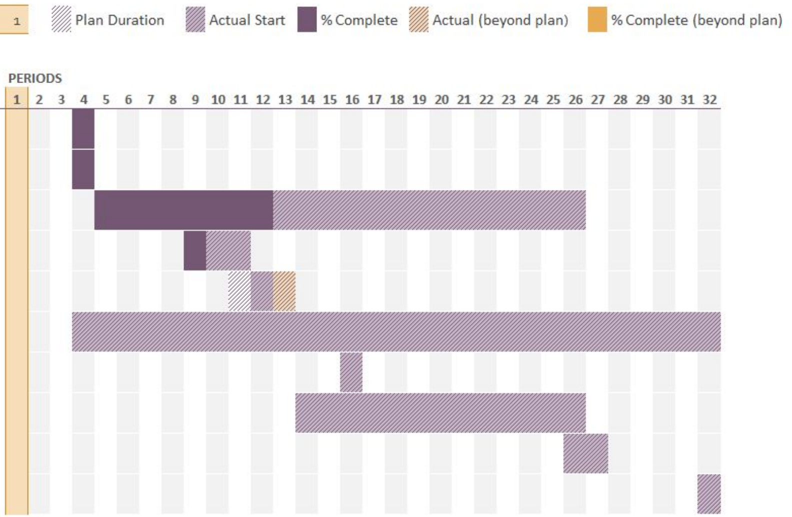
Irfan

Project Milestones & Schedule

Project Planner

Select a period to highlight at right. A legend describing the charting follows.

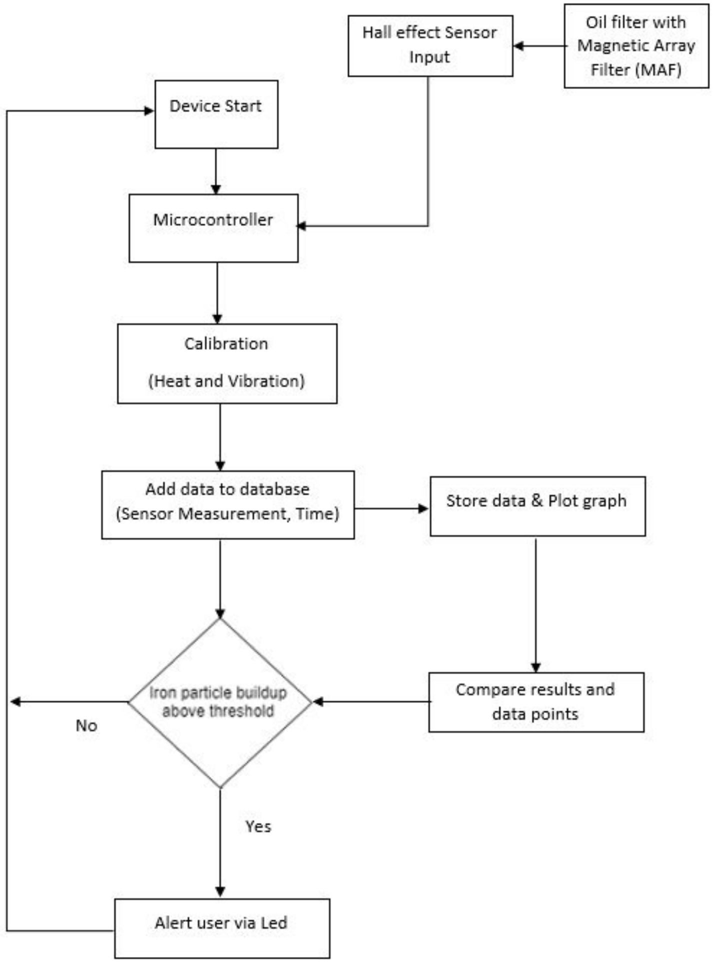
Period Highlight: 1



ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Interviewing Client (Meeting 1)	4	1	4	1	100%
Define Problem Statement	4	1	4	1	100%
Research	5	22	5	22	40%
Prototype 1	9	3	9	3	10%
Testing 1	11	2	12	2	0%
Client Meeting	4	31	4	31	0%
Presentation Fall 2019	16	1	16	1	0%
Repeat Process (Start Spring 2020)	14	13	14	13	0%
Final Testing	26	2	26	2	0%
Final Presentation Sp 2020	32	1	32	1	0%

Wei-Nee

Functional Decomposition



Aiman

IO & UI Design

Our IO is presently using .csv files and there is no live graphing. However, this can be implemented simply by using matplotlib and a python script.

Running the software can also be simplified further than the current implementation.

HW/SW/Technology Platform(s) used

Hardware:

- Arduino
- Hall sensors
- PCB

Software:

- Arduino
- Processing

Vishal

Design Tradeoffs & Innovativeness

Initial Design:

- Square PCB on top of a PVC/silicon ring
- Through-hole sensors facing inside of the collar in between the magnets
- Sensors easily damaged
- Rigid positions
- Low sensitivity towards changes in magnetic fields

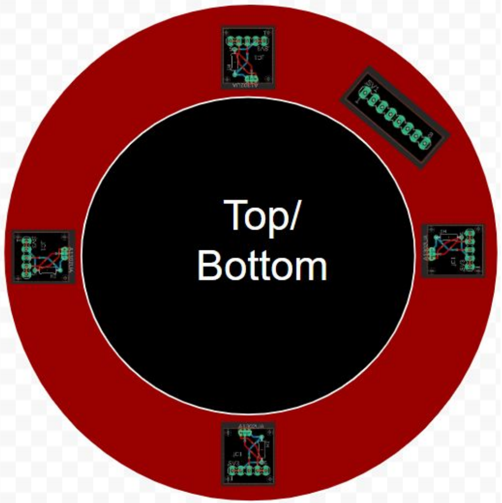


Figure 3.6: Square PCB and Output Pins on Plastic Ring Mount

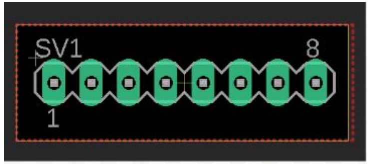


Figure 3.5: Output Pins

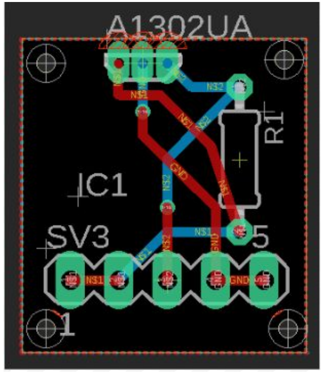


Figure 3.4: Square PCB

Aiman

Innovations for new design:

- Multiple sensors on one board
- 3D-modelled ring collar to fit with the MAF Collar
- Able to rotate to allow angle calibrations
- Fully utilizing surface-mounted sensors

Aiman

Modelling & Simulations

arduino_to_PC_serial | Processing 3.5.4

File Edit Sketch Debug Tools Help

Java ▾

arduino_to_PC_serial

```
import processing.serial.*;
import java.io.FileNotFoundException;
Serial port;
Table hallData = new Table();
int writeInterval = 500;
int counter = 1;
File chkFile;
String filePath = "/Users/aimnzulkefli/Desktop/HallData/";  //change this path to where you
String outputFile = "hallEffectData";
PrintWriter outputWriter;
void setup(){
  try{
    int checkFile = 0;
    String toCheck = filePath + outputFile + str(checkFile)+ ".csv";
    //System.out.println(toCheck);
    chkFile = new File(toCheck);
    while(chkFile.exists()){
      checkFile++;
      toCheck = filePath + outputFile + str(checkFile)+ ".csv";
      chkFile = new File(toCheck);
    }
  }
  catch(NullPointerException e){
    e.printStackTrace();
  }
  outputWriter = createWriter(chkFile);



  port = new Serial(this, Serial.list()[3], 9600); //set to portnumber of arduino
  String initialOut = "Count, Sensor 1, Sensor 2, Sensor 3, Sensor 4, Sensor 5, Sensor 6, Ti

  outputWriter.println(initialOut);
  outputWriter.flush();
}

void draw(){
  if(port.available() > 0){
    String value = port.readStringUntil('\n');
    String counterStr = str(counter) + ", ";

    if(value != null){
     System.out.println(value);
     value = value.replace("\n", "");
```

Arduino_Input | Arduino 1.8.12 (Windows Store 1.8.33.0)

File Edit Sketch Tools Help

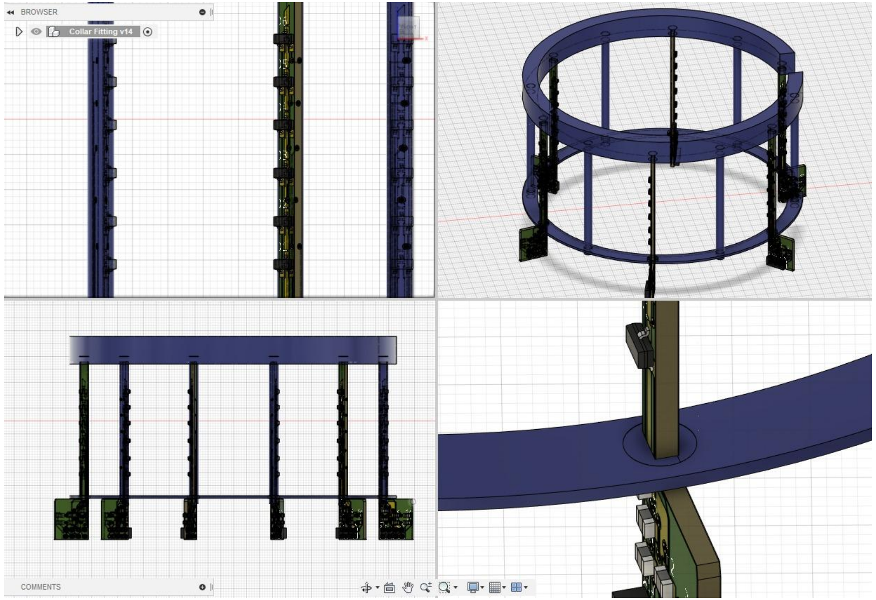
Arduino_Input

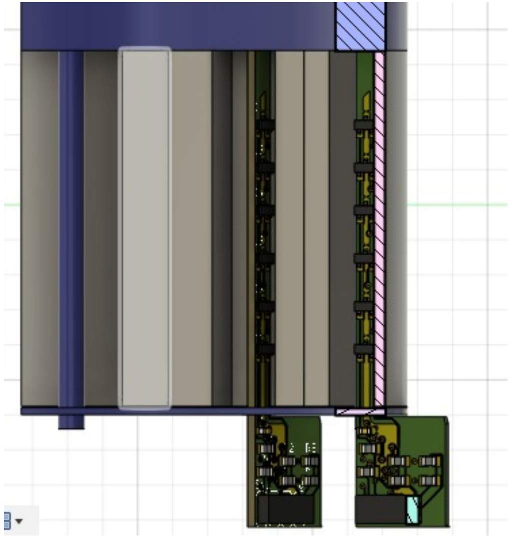
```
  int sensorPinA = 13;  //drv5057
  int sensorPinB = 12;  //drv5056
  int sensorPinC = A0;  //drv5053
  int sensorPinD = A1;
  int sensorPinE = A2;
  int sensorPinF = A3;
  String command = "";
  float highValue[] = {0,0,0,0,0,0};
  float lowValue[] = {0,0,0,0,0,0};
  float serialValue[] = {0,0,0,0,0,0};
  float  cycleValue;
void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  pinMode(sensorPinA, INPUT);
  pinMode(sensorPinB, INPUT);
}

void loop() {
  if(Serial.available() > 0){
      command = Serial.readStringUntil('\n');
      if(command.equals("CLEARFUNC")){
        }
    }
  String sendViaSerial = "";
  /*
  To read digital Hall Effect Sensor
  highValue[0] = pulseIn(sensorPinA, HIGH);
  lowValue[0] = pulseIn(sensorPinA, LOW);
  This returns time in microseconds for PWM HIGH and LOW
  Then calculate DutyCycle:
  float cycleValue = highValue[0] + lowValue[0];
  dutyValue[0] = highValue[0]/cycleValue;
  */
  highValue[0] = pulseIn(sensorPinA, HIGH);
  lowValue[0] = pulseIn(sensorPinA, LOW);
  highValue[1] = pulseIn(sensorPinB, HIGH);
  lowValue[1] = pulseIn(sensorPinB, LOW);
  serialValue[2] = analogRead(sensorPinC);
  serialValue[3] = analogRead(sensorPinD);
  serialValue[4] = analogRead(sensorPinE);
  serialValue[5] = analogRead(sensorPinF);
```

Aiman

Solutions/Innovations





Vishal

Sensor Collar Fitting

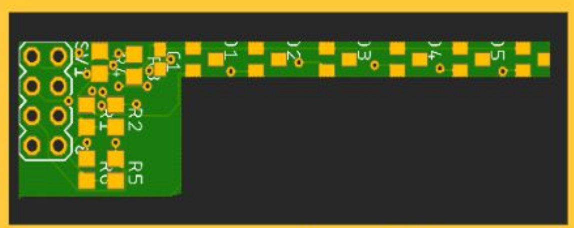
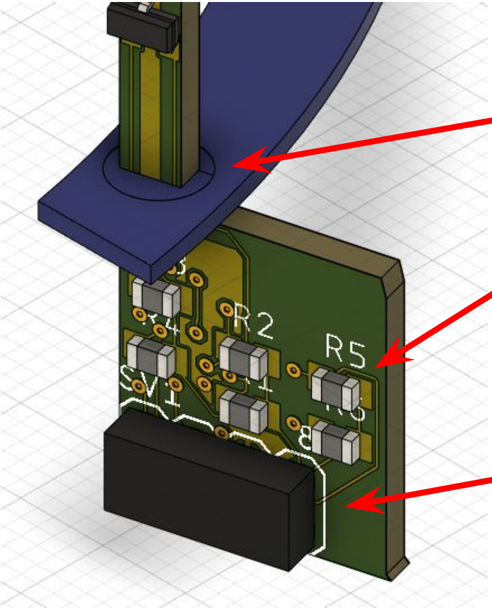


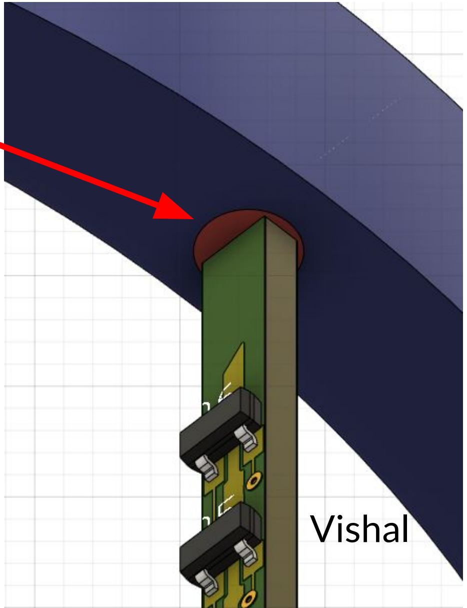
Diagram 4: Final Board design



Allows angle adjustment for calibration

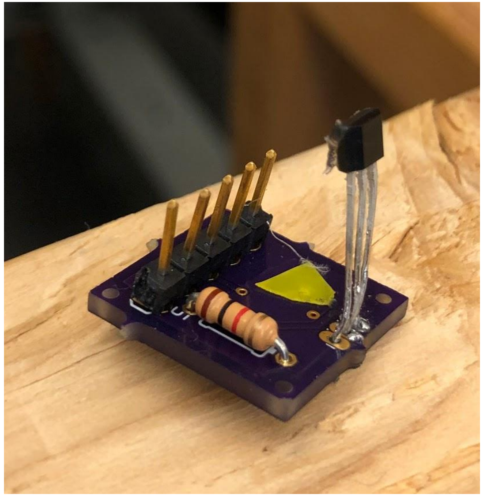
Resistors required by the sensors

Pin Connection to Arduino

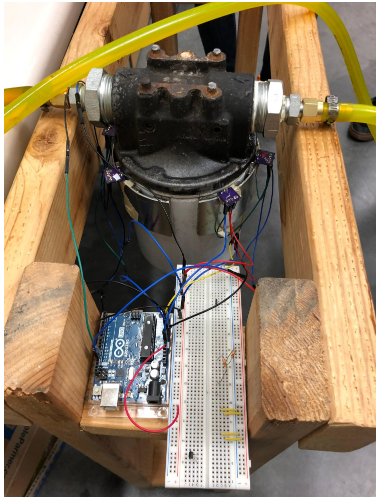


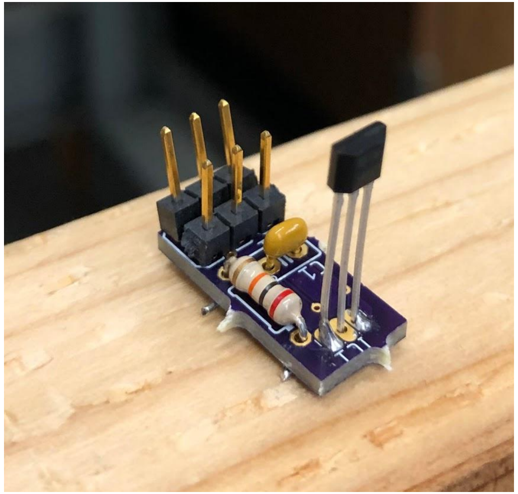
Vishal

Build Prototypes



Old PCB design 1



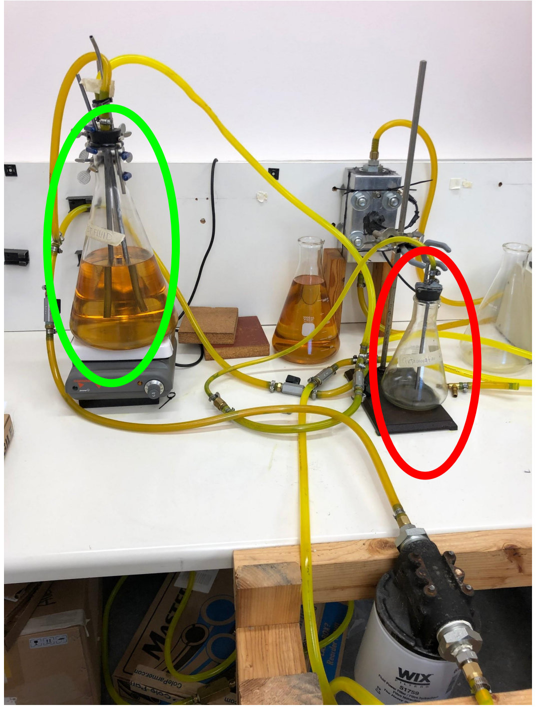


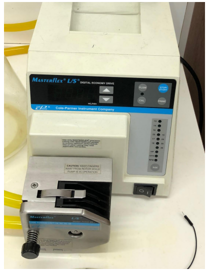
Old PCB design 2

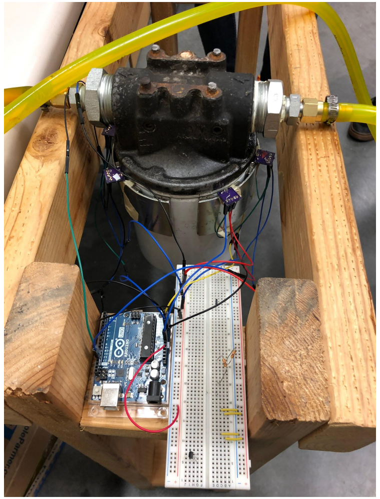
Vishal

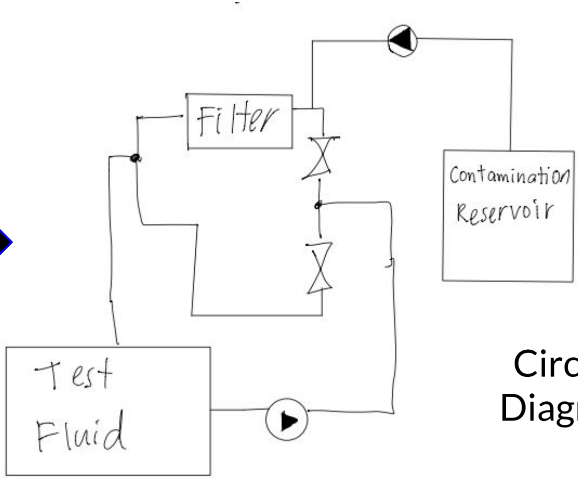
Test Plan 1

- Dr. Stewards testbench in the Fluids Lab, Sukup hall









Circuit Diagram

Lutfi

Test Plan 1

- Conducted on different days.
- Consists of 4 stages
- Every stage is taken 3 times.
- After every stage, 3g of iron particles injected

31/1	FRIDAY	lutfi, irfan, aiman	baseline	
1/2	SAT	lutfi, vishal	baseline	
2/2	SUN	lutfi	baseline	
3/2	MON	aiman	3g	ADD CONTAMINANTS 3g
4/2	TUE	irfan		
5/2	WED	weinee		
6/2	THU	vishal	6g	ADD CONTAMINANTS 3g
7/2	FRIDAY	weinee		
8/2	SAT	aiman		
9/2	SUN	vishal	9g	ADD CONTAMINANTS 3g
10/2	MON	weinee		
11/2	TUE	irfan		

Testing schedule

Lutfi

Test Plan 2

- Updated sleeve.
- Manipulating the sensor angle.
- Multiple sensors on a single PCB.
- Various Sensor sensitivity.

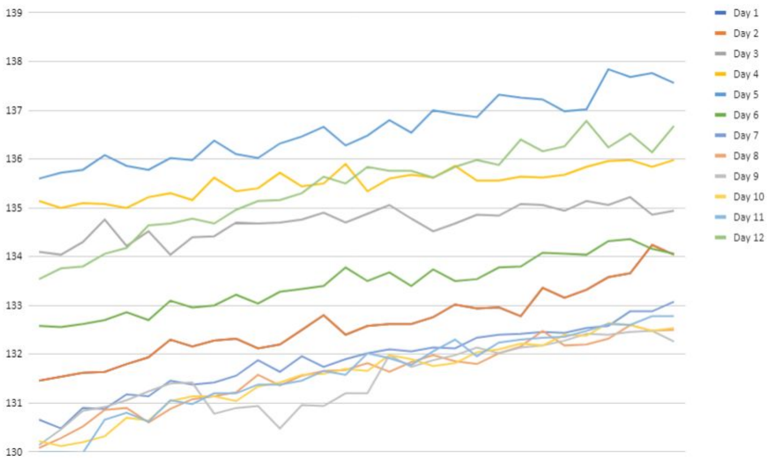
Accumulation of iron particles on mixer stick

Lutfi

Test Results & Evaluations

Our initial test results were inconclusive but we do have made some changes in terms of calibration.

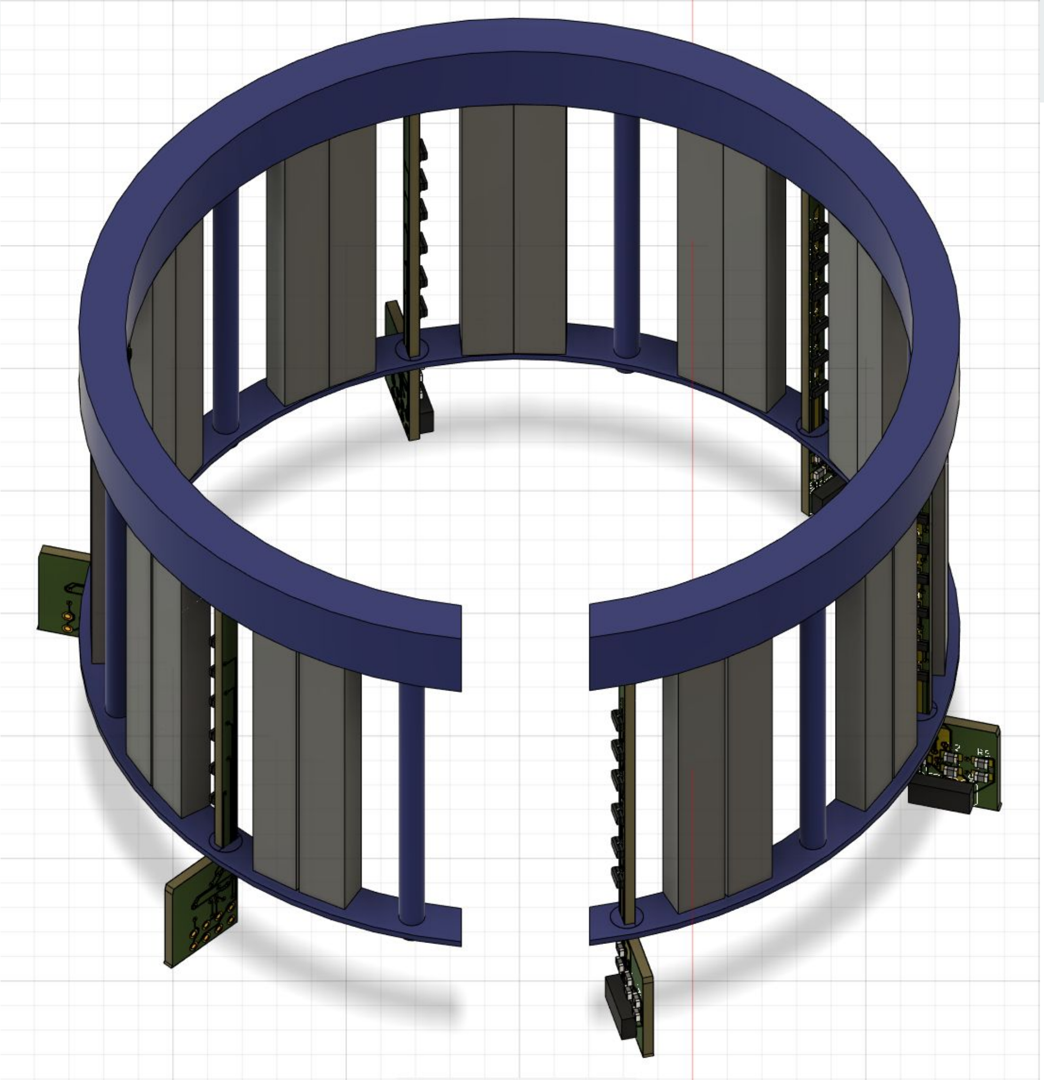
We believe our results were inconclusive due to miscalibrated sensors and sensor choice.



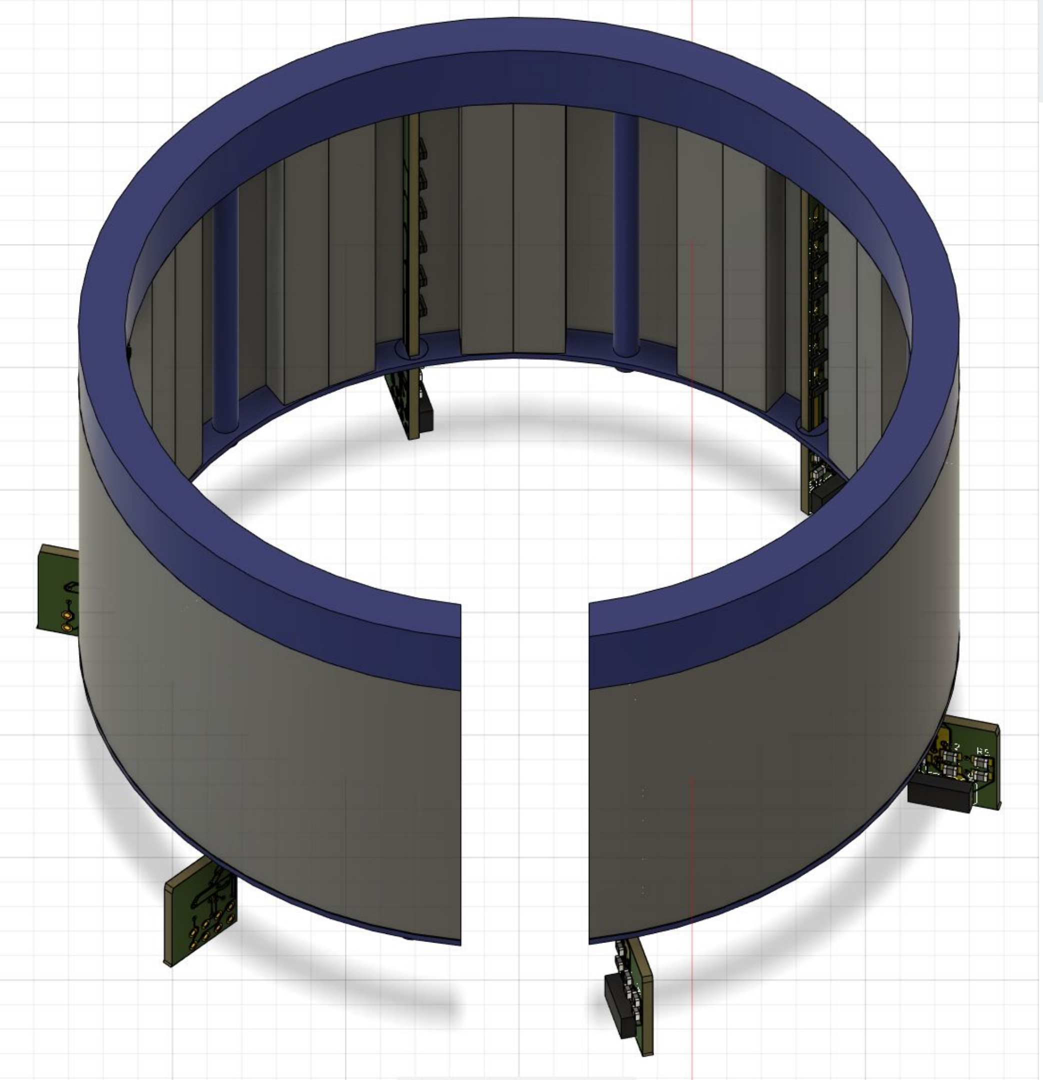
Irfan

Prototype Implementations or basic building block implementations (and applicable results)

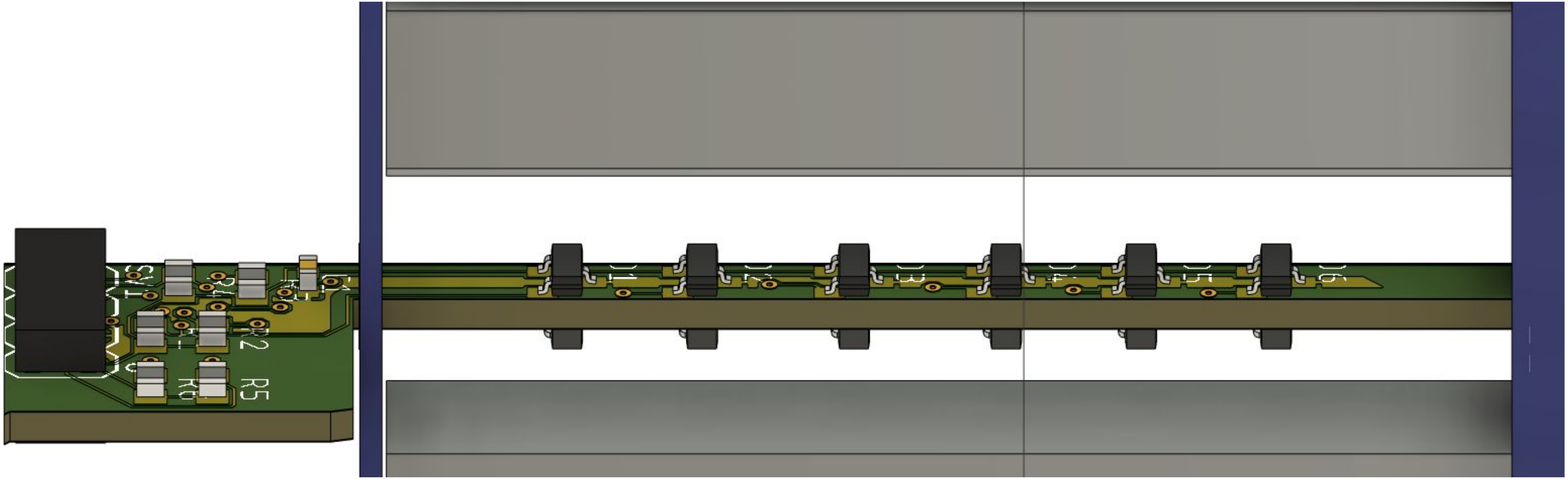
Irfan



Irfan



Irfan



Irfan

Engineering Standards and Design Practices

- IEEE 1522-2004 - IEEE Standard for Testability and Diagnosability
- IEEE 1413-2010 - IEEE Standard Framework for Reliability Prediction of Hardware
- IEEE 1641-2010 - IEEE Standard for Signal and Test Definition

Vishal

Conclusion & Lessons Learned

- Datasheets contain a multitude of useful information that may be useful to the project, so skimming through it to find out
- Always prepare contingency plans