



PSG INSTITUTE OF TECHNOLOGY AND APPLIED RESEARCH

Neelambur, Coimbatore – 641 062

Smart Drainage, Effluent and Sewage Pumping System



TEAM:

PSG iTech – 2

CATEGORY:

Smart
Machines

Team Details



Participant Name	CT /DT Number	Role (Team Leader / Member)	Bachelors Discipline	Expected Year of Passing	Gender
Shravan S	CT20182402848	Team Leader	Electrical and Electronics Engineering	2020	M
Raagul A S	CT20182408578	Team Member	Mechanical Engineering	2020	M
Anirudh P S	CT20182401507	Team Member	Electronics and Communication Engineering	2020	M
Ramprakash V	CT20182408971	Team Member	Computer Science Engineering	2020	M



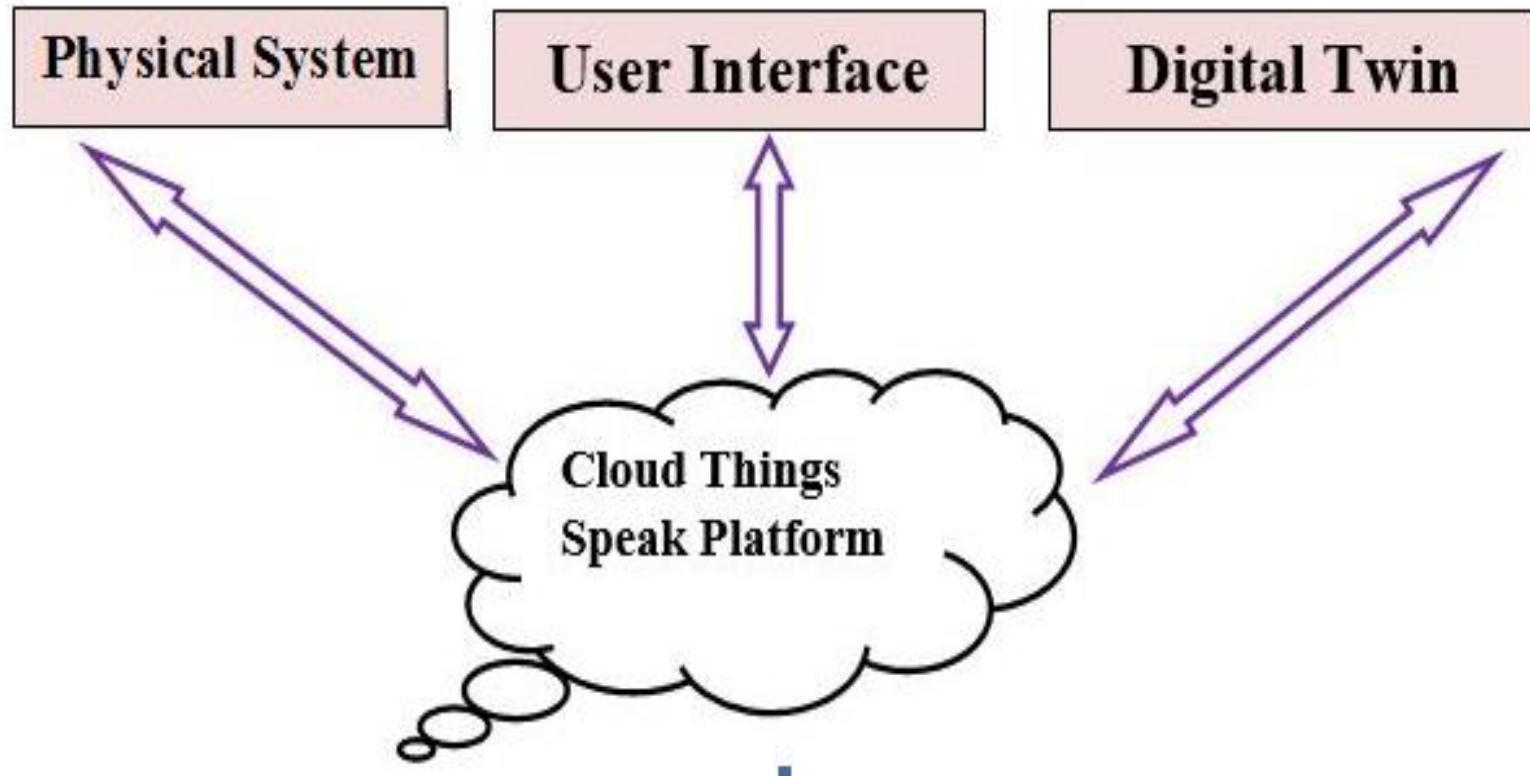
- Pumping of drainage, effluent and sewage is different from pumping of water due to the polluted nature of the materials.
- The pollutants are in the form of suspended solids and floating solids. Hence, the **load on the pump varies widely.**
- The proposed system will monitor single phasing fault, stator temperature, vibration, and ambient temperature, and will **dynamically control the speed of the pump and ensure sustained reliable performance of the pump.**



- Intelligence, by way of machine learning, is built into the system using **mathematical expressions that characterize the physical system.**
- The users can **query the Digital Twin** to know the historical performance, and current operating conditions.
- The proposed system can **trigger alarms as early warnings, and also make predictions** about possible system anomalies, if and when they occur



- Intelligence, by way of machine learning, is built into the system using **mathematical expressions that characterize the physical system, and data analytics.**
- The users can **query the digital twin** to know the historical performance, and current operating conditions.
- The proposed system can **trigger alarms as early warnings, and also make predictions** about possible system anomalies, if and when they occur

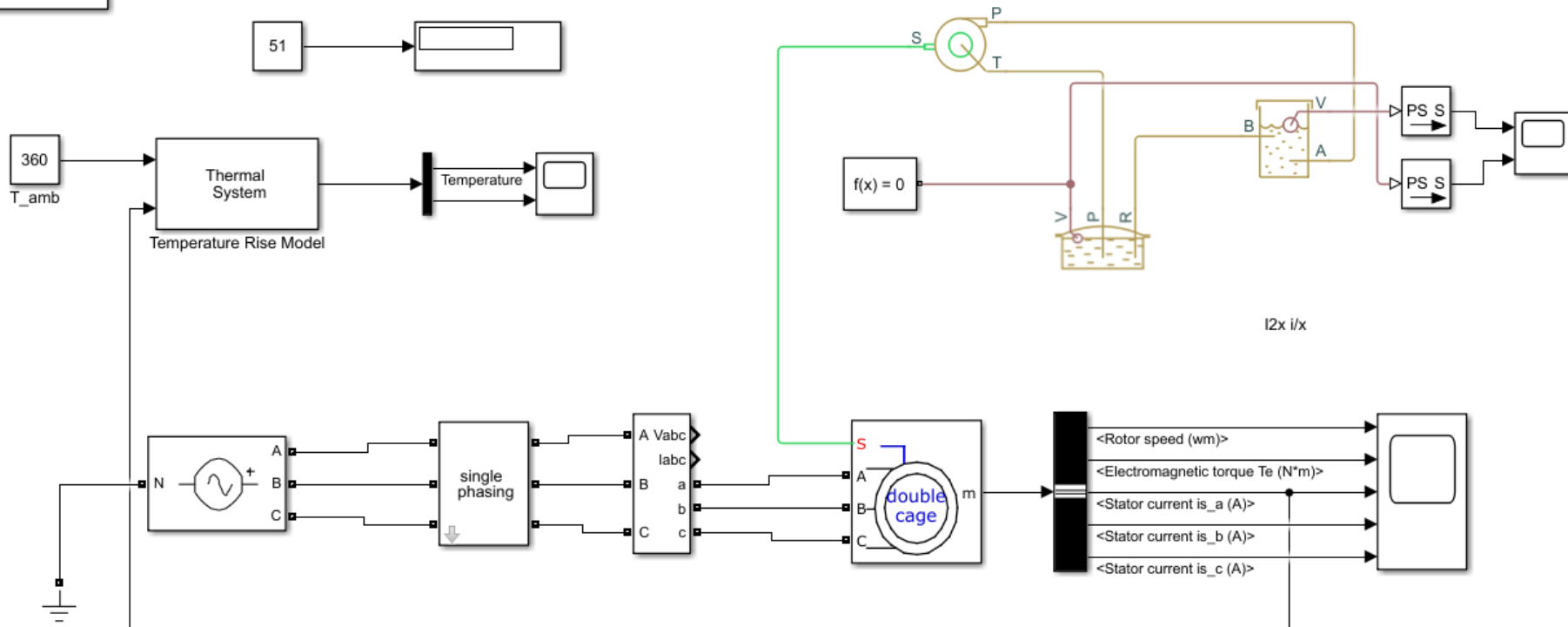




Digital Twin

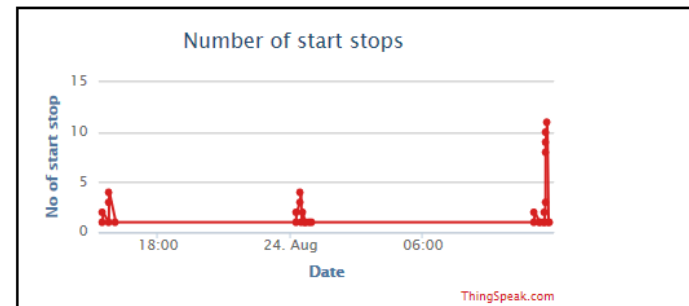
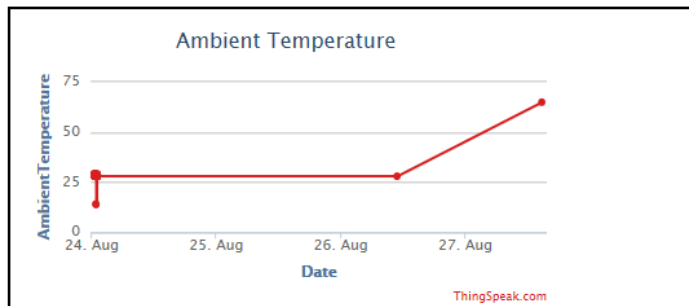
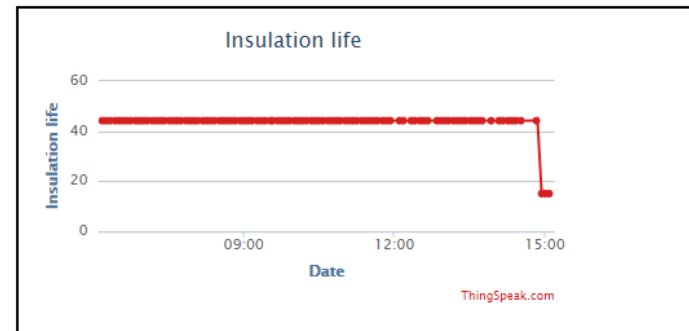
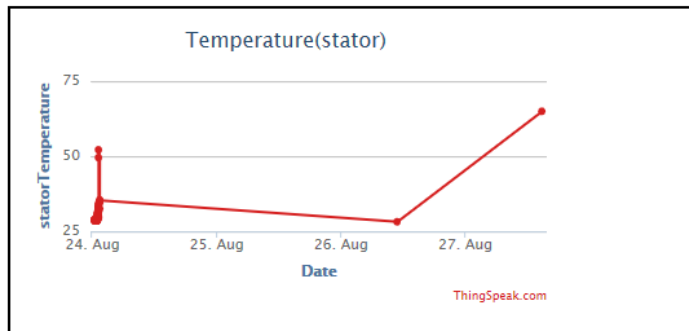


Continuous





SMART DRAINAGE EFFLUENT AND SEWAGE PUMPING SYSTEM



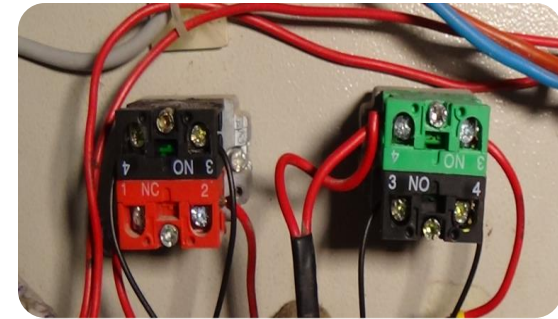


SYSTEM OPERATION

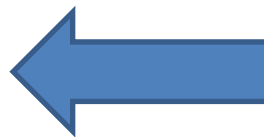
Starting The Digital Twin



Start push button is pressed



Auxiliary contactors gets actuated



Starting The Digital Twin

```
pi@raspberrypi: ~/enginx/adx1345-python
File Edit View Help
pi@raspberrypi: ~/enginx/adx1345-python $ python start2.py
start2.py:16: SyntaxWarning: name 'flag' is assigned to before global declaration
    global flag
start2.py:24: SyntaxWarning: name 'flag' is assigned to before global declaration
    global flag
start2.py:26: SyntaxWarning: name 'counts' is assigned to before global declaration
    global counts
start
```

```
Command Window
>> cloud
Warning: Cannot query I2C bus speed.
> In raspi/getAvailablePeripherals (line 929)
   In raspi (line 247)
   In cloud (line 6)
motor started0.1417180.1507130.1505130.141550
3.767813
```

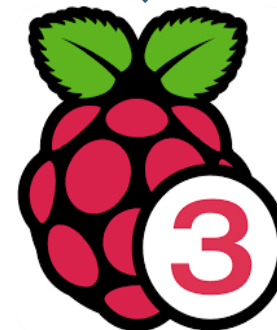
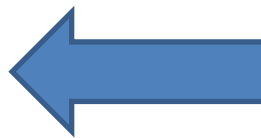
Stopping The Digital Twin



Stop push button is pressed



Auxiliary contactors get actuated



Stopping The Digital Twin

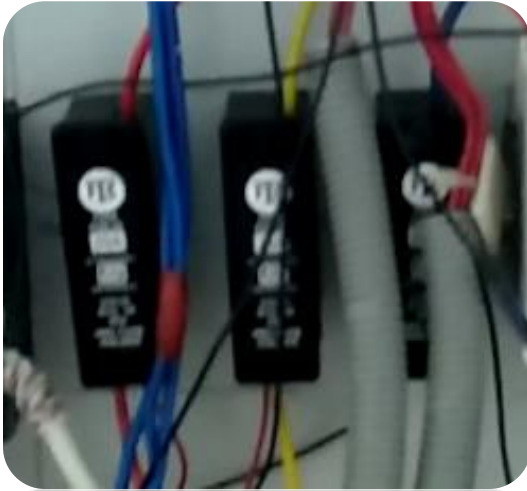
```
raspberrypi: /engine/bin/100 python start2.py
start2.py:16: SyntaxWarning: name 'flag' is assigned to before global declaration
    global flag
start2.py:24: SyntaxWarning: name 'flag' is assigned to before global declaration
    global flag
start2.py:26: SyntaxWarning: name 'counts' is assigned to before global declaration
    global counts
start2.py:8: RuntimeWarning: This channel is already in use, continuing anyway.
Use GPIO.setwarnings(False) to disable warnings.
    GPIO.setup(17,GPIO.OUT)
start
stop
1
start
stop
2
```

Command Window

```
>>
>>
>>
>>
/x >> Motor stopped
```

SINGLE PHASING FAULT

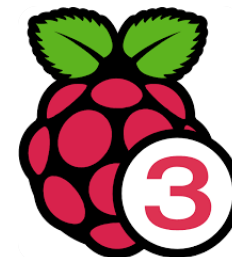
Single Phasing Fault



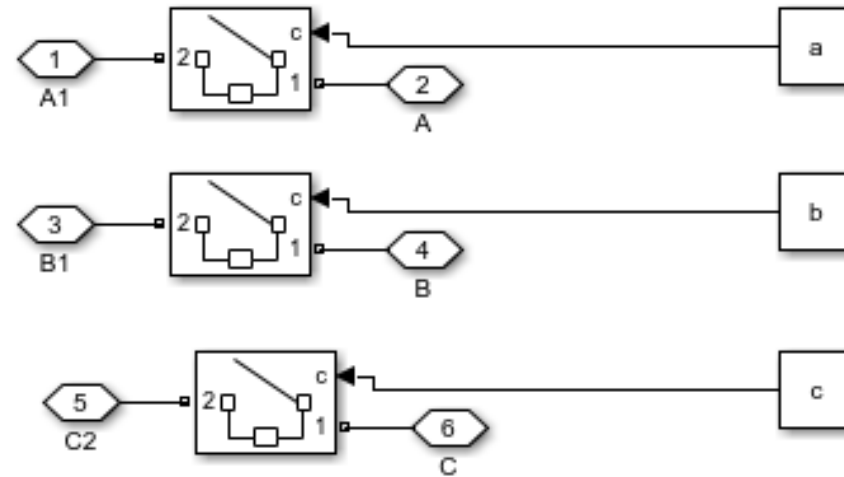
Electrical Fuse



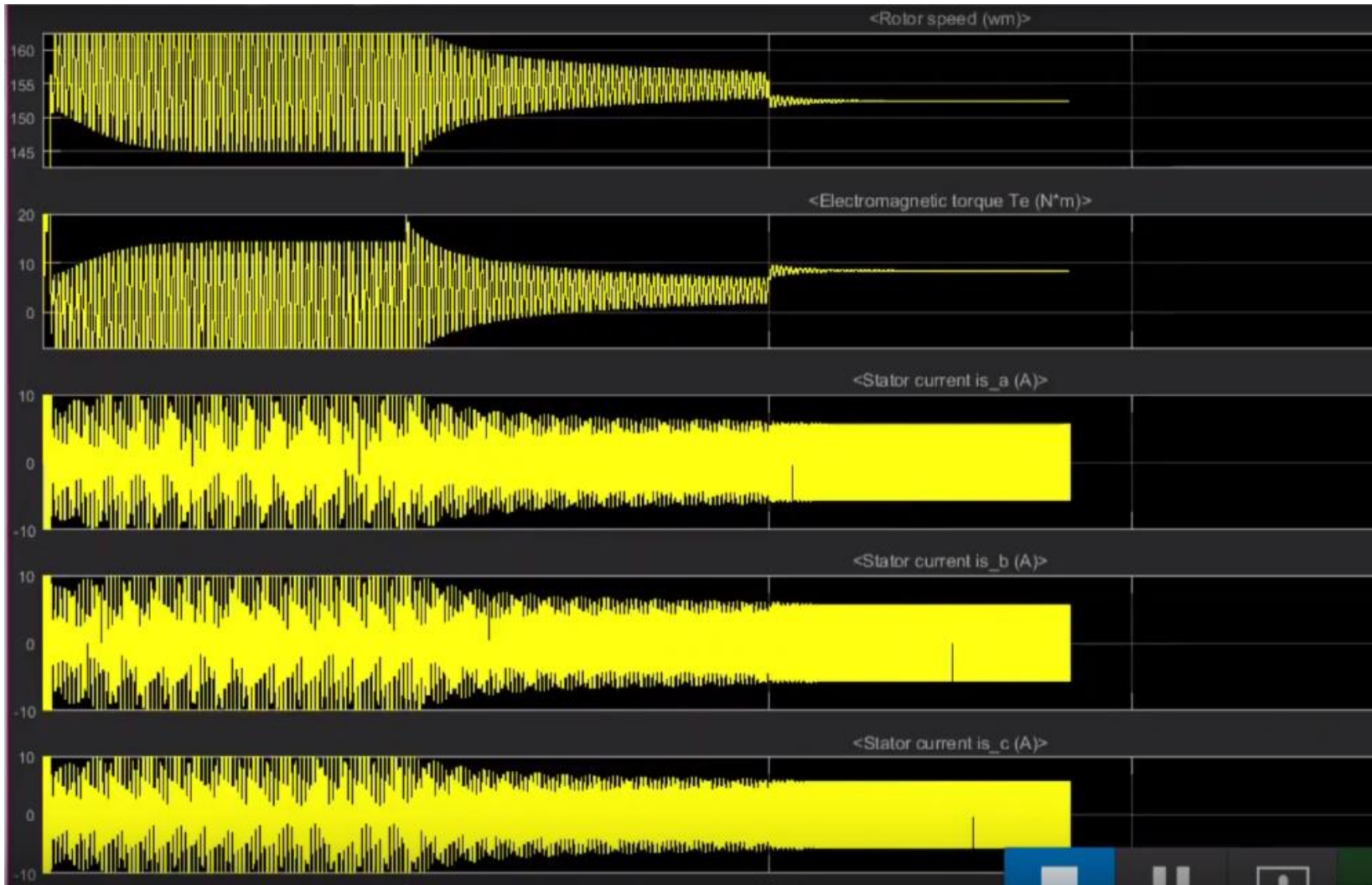
Energy Meter



Single Phasing Fault-Twin



Electrical Parameters Waveform



Single Phasing Fault-Waveform



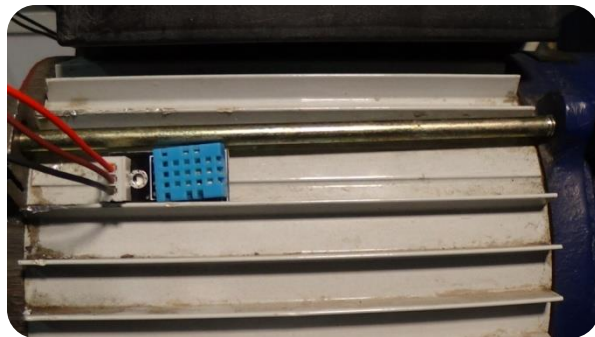


THERMAL SUBSYSTEM

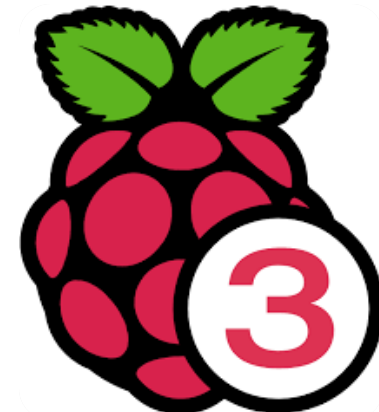
Insulation Life Estimation



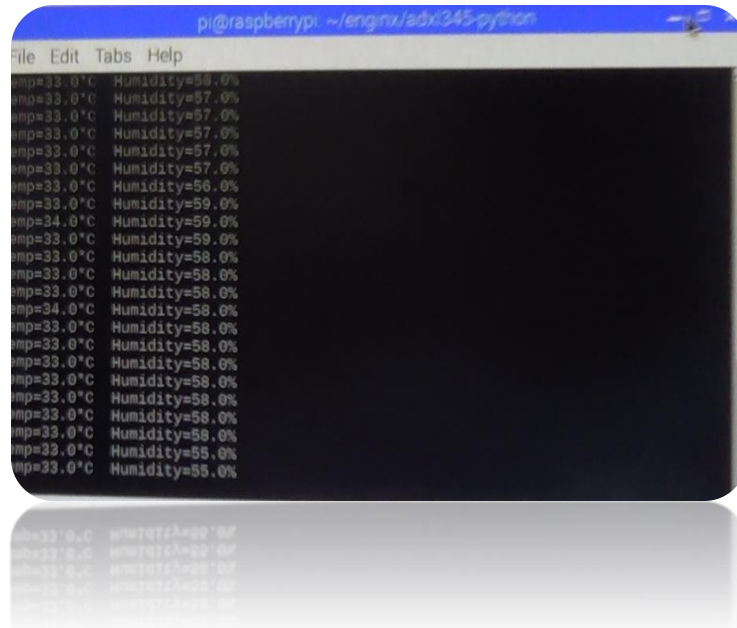
Load current data from
Energy Meter



Ambient Temperature
from DHT 11



Insulation Life Estimation



```
pi@raspberrypi: ~/enginx/adx345-python
File Edit Tabs Help
temp=33.0°C Humidity=58.0%
temp=33.0°C Humidity=57.0%
temp=33.0°C Humidity=57.0%
temp=33.0°C Humidity=57.0%
temp=33.0°C Humidity=57.0%
temp=33.0°C Humidity=57.0%
temp=33.0°C Humidity=56.0%
temp=33.0°C Humidity=59.0%
temp=34.0°C Humidity=59.0%
temp=33.0°C Humidity=59.0%
temp=33.0°C Humidity=58.0%
temp=33.0°C Humidity=58.0%
temp=33.0°C Humidity=58.0%
temp=33.0°C Humidity=58.0%
temp=34.0°C Humidity=58.0%
temp=33.0°C Humidity=58.0%
temp=33.0°C Humidity=58.0%
temp=33.0°C Humidity=58.0%
temp=33.0°C Humidity=58.0%
temp=33.0°C Humidity=58.0%
temp=33.0°C Humidity=58.0%
temp=33.0°C Humidity=58.0%
temp=33.0°C Humidity=58.0%
temp=33.0°C Humidity=55.0%
temp=33.0°C Humidity=55.0%
```



Ambient Temperature readings in Raspberry Pi Screen

Ambient Temperature Readings Being Plotted In ThingSpeak

Insulation Life Estimation




ThingSpeak Channels Apps Blog Support Sign In Sign Up


Apps

ThingSpeak channels store data. Upload data from the web or send data from devices to a ThingSpeak channel. Use these apps to transform and visualize data or trigger an action. See [Tutorial: ThingSpeak and MATLAB](#) to create a channel.


Analytics



MATLAB Analysis
Explore and transform data.



MATLAB Visualizations
Visualize data in MATLAB plots.



Plugins
Display data in gauges, charts, or custom plugins.

Ambient Temperature Readings Being Plotted In ThingSpeak



Load Current Data From MATLAB

Insulation Lifetime Estimation Using MATLAB Analytics in ThingSpeak platform



Algorithms Behind Insulation Life Estimation:

- Montsinger Rule
- Arrhenius Equation
- Final consolidated equations:

1. Hotspot Temperature =[Loss Factor* Allowable temperature rise]+Ambient Temperature

$$2. L_x = L_{100} * 2 \exp \left[\frac{T_c - T_x}{HIC} \right]$$



Code Snippets:

```
1 clc;
2 % Parameter Declarations
3 Class=[105 130 155 180];
4 HIC=[14 11 9.3 8];
5 Load_Factor=50:1:150;
6 [r c]=size(Load_Factor);
7 Loss_Factor=zeros(1,c);
8 for i=1:1:c
9     Loss_Factor(i)=0.012*Load_Factor(i)*Load_Factor(i)-(0.83*Load_Factor(i))+65;
10 end
11 Loss_Factor;
12 %%
13 Channel_ID=559226;
14 Channel_ID2=562478;
15 [Torque,timestamp]=thingSpeakRead(Channel_ID,'Fields',1,'NumPoints',10)
16 [rr cc]=size(Torque);
17 Torque_Tot=0;
18 count=0;
19 for i=1:1:rr
20     temp=Torque(i);
21     if isnan(temp)==0
22         Torque_Tot=Torque_Tot+temp;
23         count=count+1;
24     end
25 end
26 Torque_Tot=Torque_Tot;
27 Temp_amb=thingSpeakRead(Channel_ID2,'ReadKey','PBLRW9SSRWYQI66Z','Fields',1);
28 Time_Period=20; % Current Load Duty time
```

```
28 Time_Period=20; % Current Load Duty time
29 Base_Load_Time=24;
30 LV_PU = Time_Period/Base_Load_Time;
31 Load_Tot= ceil(sum(Torque_Tot)/(LV_PU+count-1))
32 % Getting Class Insulation
33 %Ins_Class_Cell=thingSpeakRead(Channel_ID,'Fields',3,'OutputFormat','table');
34 %Ins_Class=Ins_Class_Cell.InsulationClass{1};
35 Ins_Class='B';
36
37 if Ins_Class=='A'
38     c=Class(1);
39     h=HIC(1);
40 end
41 if Ins_Class=='B'
42     c=Class(2);
43     h=HIC(2);
44 end
45 if Ins_Class=='F'
46     c=Class(3);
47     h=HIC(3);
48 end
49 if Ins_Class=='H'
50     c=Class(4);
51     h=HIC(4);
52 end
53 Inter_Load=Load_Factor-Load_Tot;
54 z=find(Inter_Load==0)
55 F=Loss_Factor(z)/100;
56 fprintf('Loss Factor F: %f\n',F);
57 T_hot=F*(c-Temp_amb)+Temp_amb;
58 fprintf('Hot spot Tx: %f\n',T_hot);
59 Lx=100*(2^((c-T_hot)/h));
60 E_year=((Lx*20000)/(100*24*365));
```

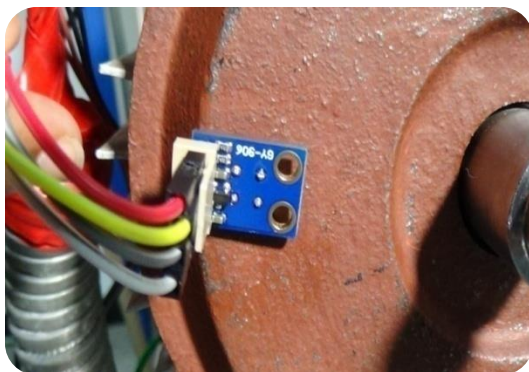
```
61 E_hours=E_year*24*365;
62 Elapsed=(LV_PU+count-1)*24;
63 E_year_rem=(E_hours-Elapsed)/(365*24);
64 fprintf('Expected life in year: %f \nDays: %f \nHours: %f\n',E_year_rem,E_year_rem*365,E_y
65 thingSpeakWrite(559243,E_year,'Fields',1,'WriteKey','SRHS7G3QM6EE7A0H');
66 if E_year*365 <100
67     fprintf('Please Check for Insulation Replacement \n');
68     thingSpeakWrite(563794,1,'WriteKey','T07N0YH6P6LL9WDB');
69 else
70     thingSpeakWrite(563794,0,'WriteKey','T07N0YH6P6LL9WDB');
71 end
```



```
Loss Factor F: 0.542880  
Hot spot Tx: 88.859200  
Expected life in year: 30.493791  
Days: 11130.233571  
Hours: 267125.605715
```

Output Window Displaying The Estimated Insulation Life

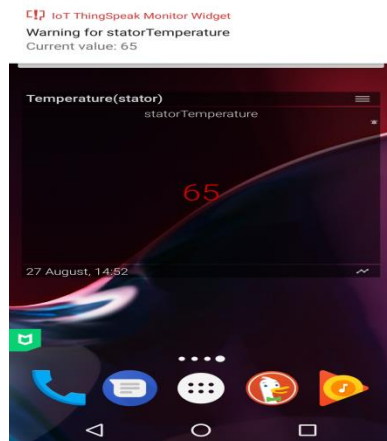
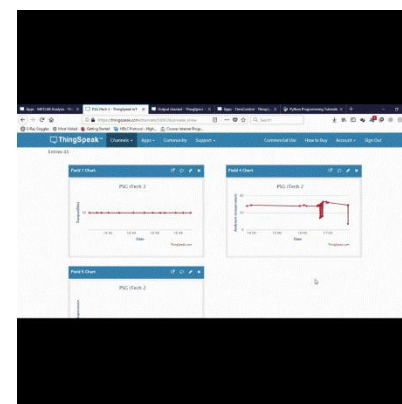
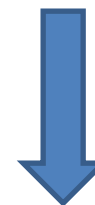
Monitoring Temperature Rise In Stator Winding



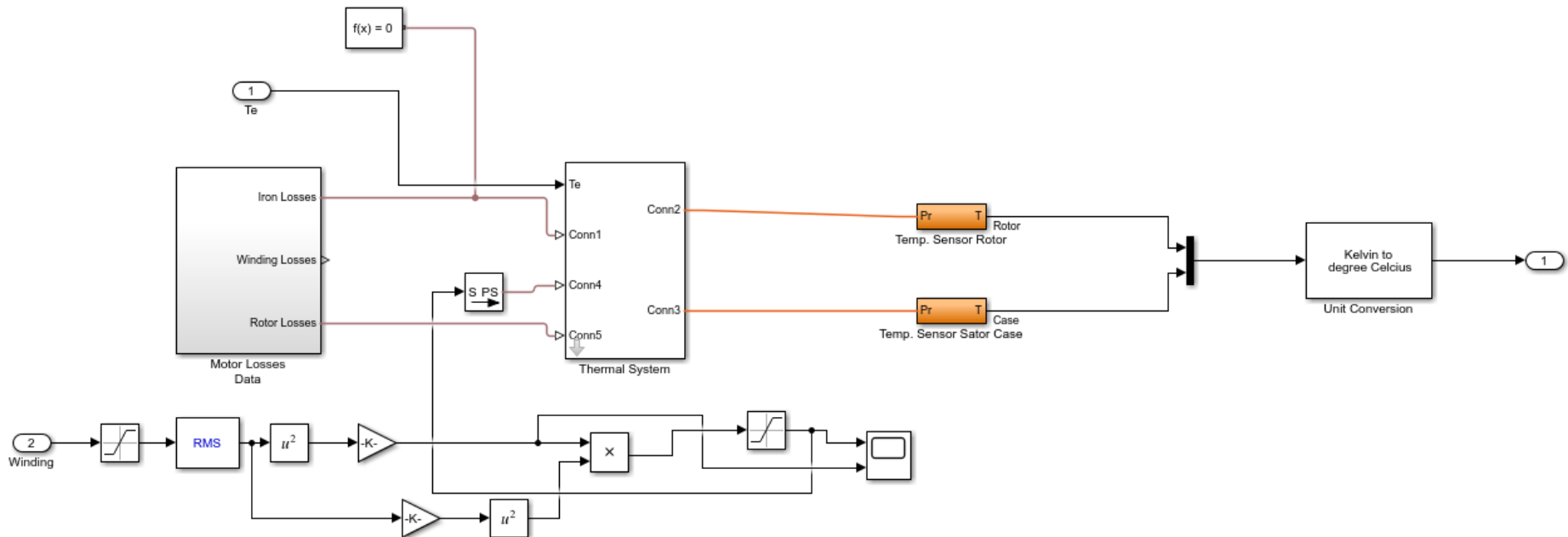
Stator Winding Temperature
being detected by MLX90614



 **ThingSpeak**



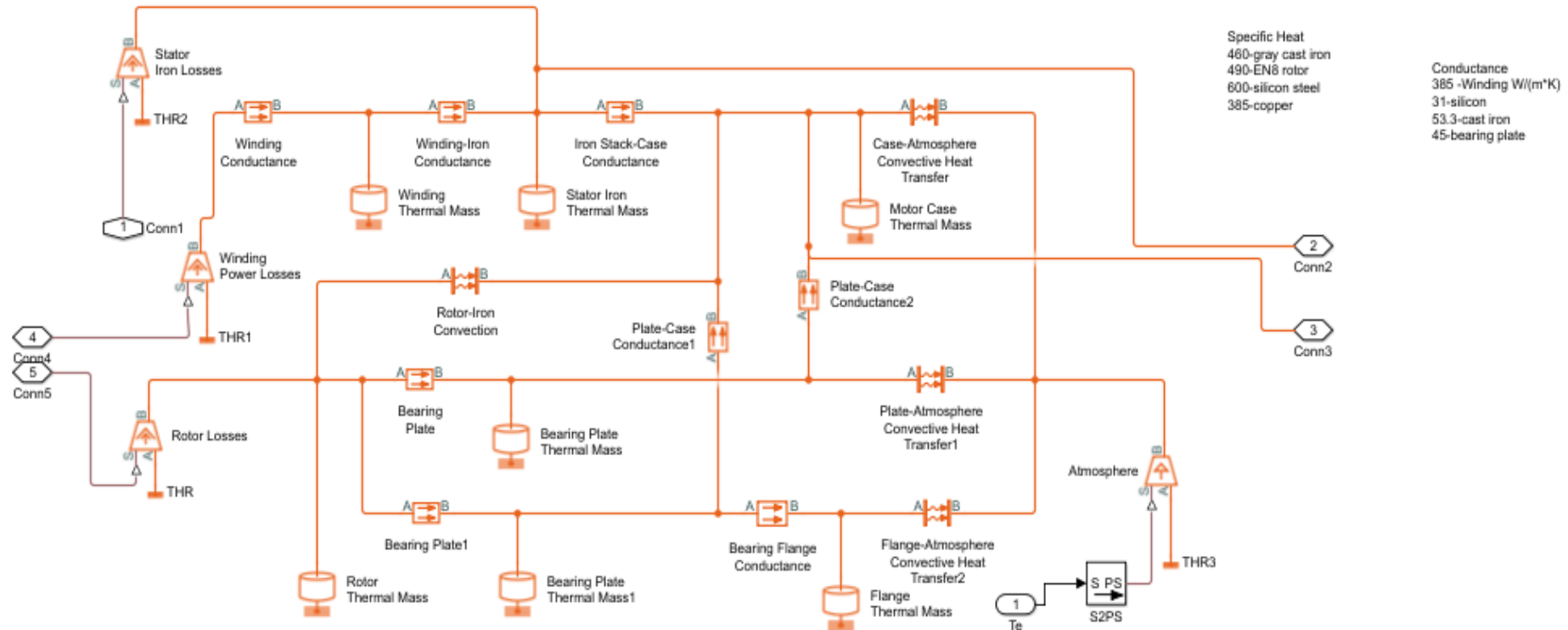
Thermal Sub System-Digital Twin



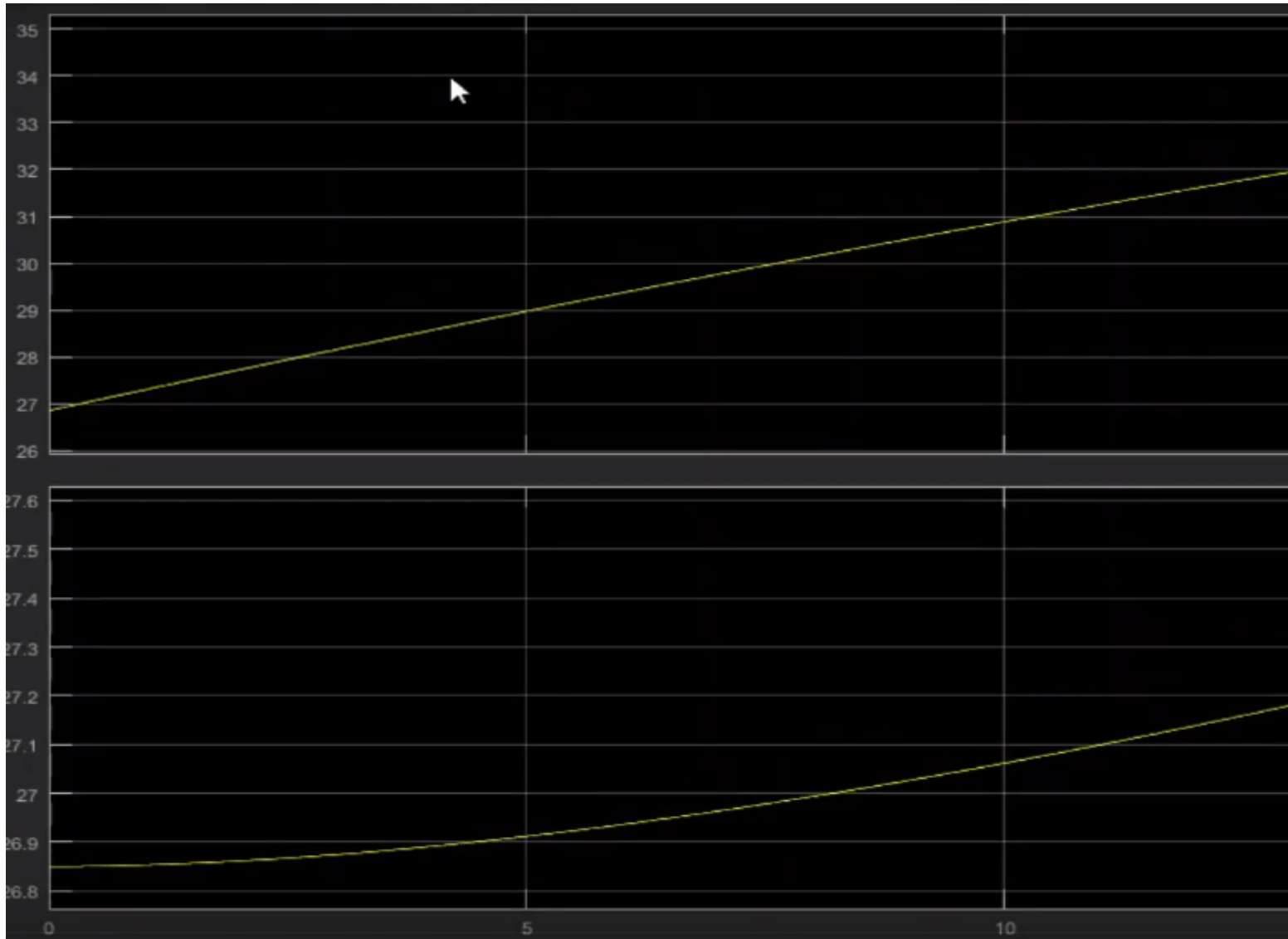
Heat Transfer-Digital Twin



TATA CONSULTANCY SERVICES



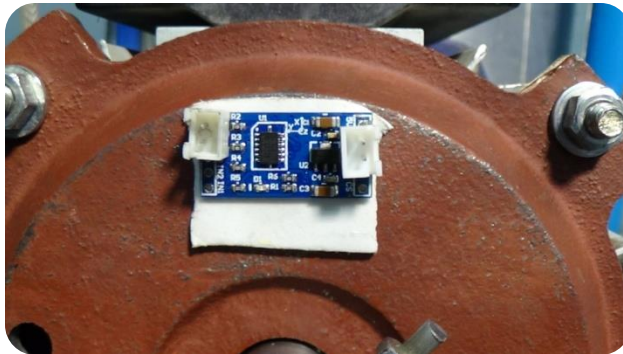
Temperature Rise Waveform



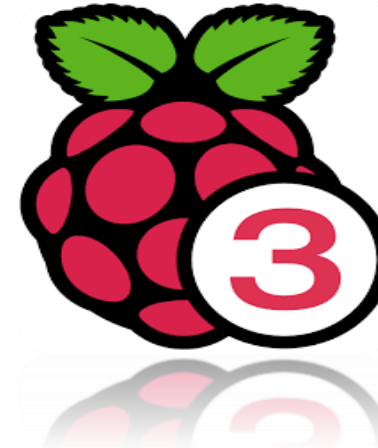


VIBRATION ANALYSIS

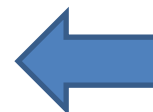
Vibration Analysis



ADXL345
mounted on the
casing

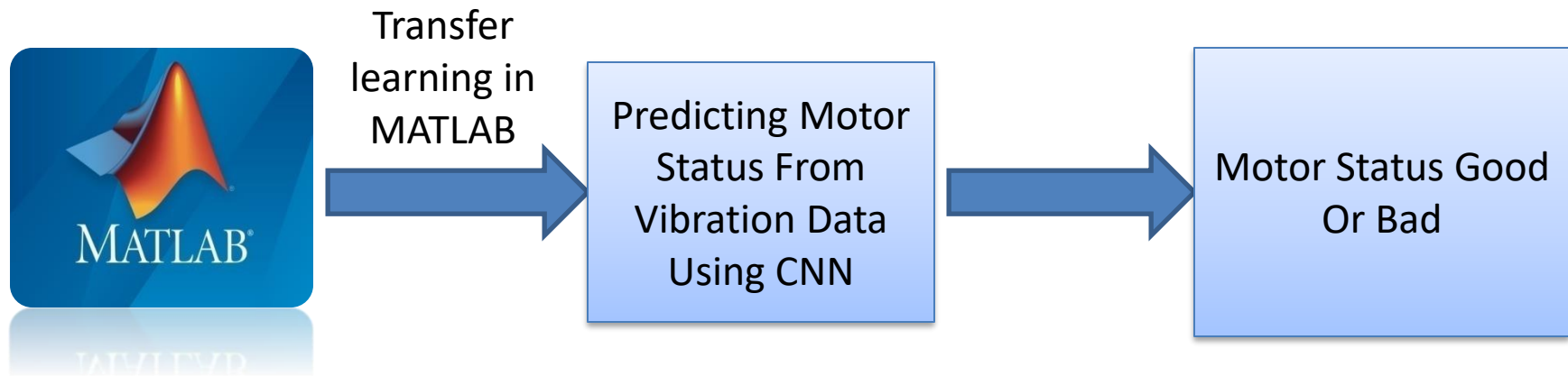


10 May 2024

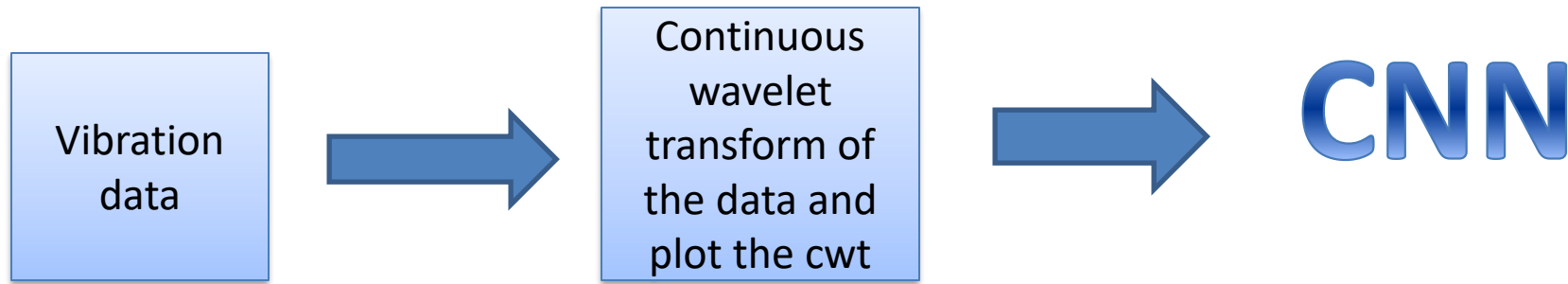


EngiNx 2018

```
pi@raspberrypi: ~/enginx/adxl345-python
y = -1.0486
z = -0.2086
ADXL345 on address 0x53:
x = 2.0446
y = -1.7086
z = 2.0446
mount the sensor properly
ADXL345 on address 0x53:
x = 1.7926
y = -1.1286
z = 2.0446
ADXL345 on address 0x53:
x = 1.2806
y = 0.7126
z = 2.0446
ADXL345 on address 0x53:
x = 2.0446
y = -2.0486
z = 1.9446
ADXL345 on address 0x53:
x = 2.0446
y = -2.0486
z = 1.3706
```



VIBRATION ANALYSIS:



VIBRATION ANALYSIS:

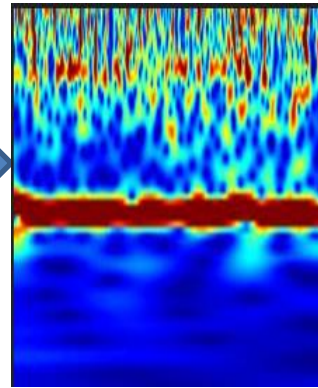


TATA CONSULTANCY SERVICES

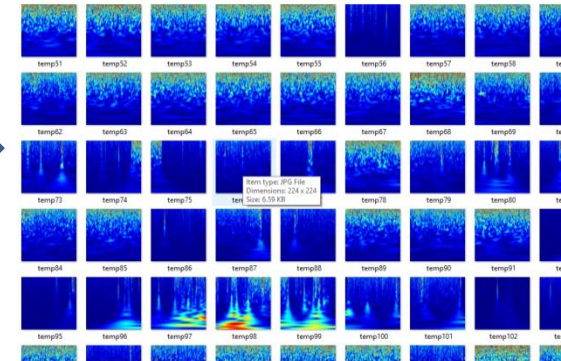


IMPORT		VIEW	
Delimited		Column delimiters: Comma	Range: A2:E
Fixed Width		Delimiter Options	Variable Names Row: 1
DELIMITERS			
vibration.csv			
vibration			
time	axis	axis	axis
datetime	Number	Number	Number
1	22:16:43	-0.048	-0.24
2	22:16:43	2.044	-2.048
3	22:16:43	2.044	-1.48
4	22:16:43	2.044	-1.696
5	22:16:43	2.044	-2.048
6	22:16:43	0.6	-0.352
7	22:16:43	2.044	-2.048
8	22:16:43	2.044	-1.536
9	22:16:43	1.456	0.032
10	22:16:43	1.848	-1.408
11	22:16:43	2.044	-2.048
12	22:16:43	1.544	0.408
13	22:16:43	2.044	-2.048
14	22:16:43	1.568	-2.048
15	22:16:43	2.044	0.152
16	22:16:43	2.044	-1.992
17	22:16:43	2.044	-1.688
18	22:16:43	1.792	-0.872
19	22:16:43	2.044	-1.128
20	22:16:43	2.016	-0.856
21	22:16:43	2.044	-1.872
22	22:16:43	-0.24	0.16
23	22:16:43		-0.004

Vibration data



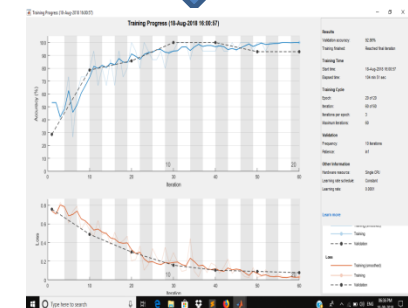
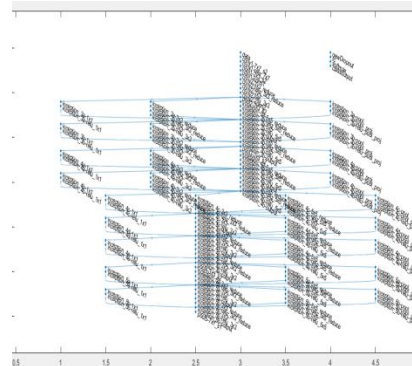
CWT of the data and plot the CWT



Create image data store

GOOD
Or
BAD

Motor status



Training the CNN



Code snippets for training

MATLAB R2017b

HOME PLOTS APPS EDITOR PUBLISH VIEW

FILE NAVIGATE EDIT BREAKPOINTS RUN

Current Folder: C:\Users\Ramprakash\Desktop\googlenet

Editor: C:\Users\Ramprakash\Desktop\googlenet\trainnetwork.m

```
1 lgraph = removeLayers(lgraph,{'pool5-drop_7x7_s1','loss=classifier','prob','output'});
2
3 numClasses = numel(categories(imgsTrain.Labels));
4 newLayers = [
5     dropoutLayer(0.6,'Name','newDropout')
6     fullyConnectedLayer(numClasses,'Name','fc','WeightLearnRateFactor',5,'BiasLearnRateFactor',5)
7     softmaxLayer('Name','softmax')
8     classificationLayer('Name','classoutput')];
9 lgraph = addLayers(lgraph,newLayers);
10
11 lgraph = connectLayers(lgraph,'pool5-7x7_s1','newDropout');
12 inputSize = net.Layers(1).InputSize;
13 options = trainingOptions('sgdm',...
14     'MiniBatchSize',15,...
15     'MaxEpochs',20,...
16     'InitialLearnRate',1e-4,...
17     'ValidationData',imgsValidation,...
18     'ValidationFrequency',10,...
19     'ValidationPatience',Inf,...
20     'Verbose',1,...
21     'ExecutionEnvironment','cpu',...
22     'Plots','training-progress');
23 rng default
24 trainedGN2 = trainNetwork(imgsTrain,lgraph,options);
```

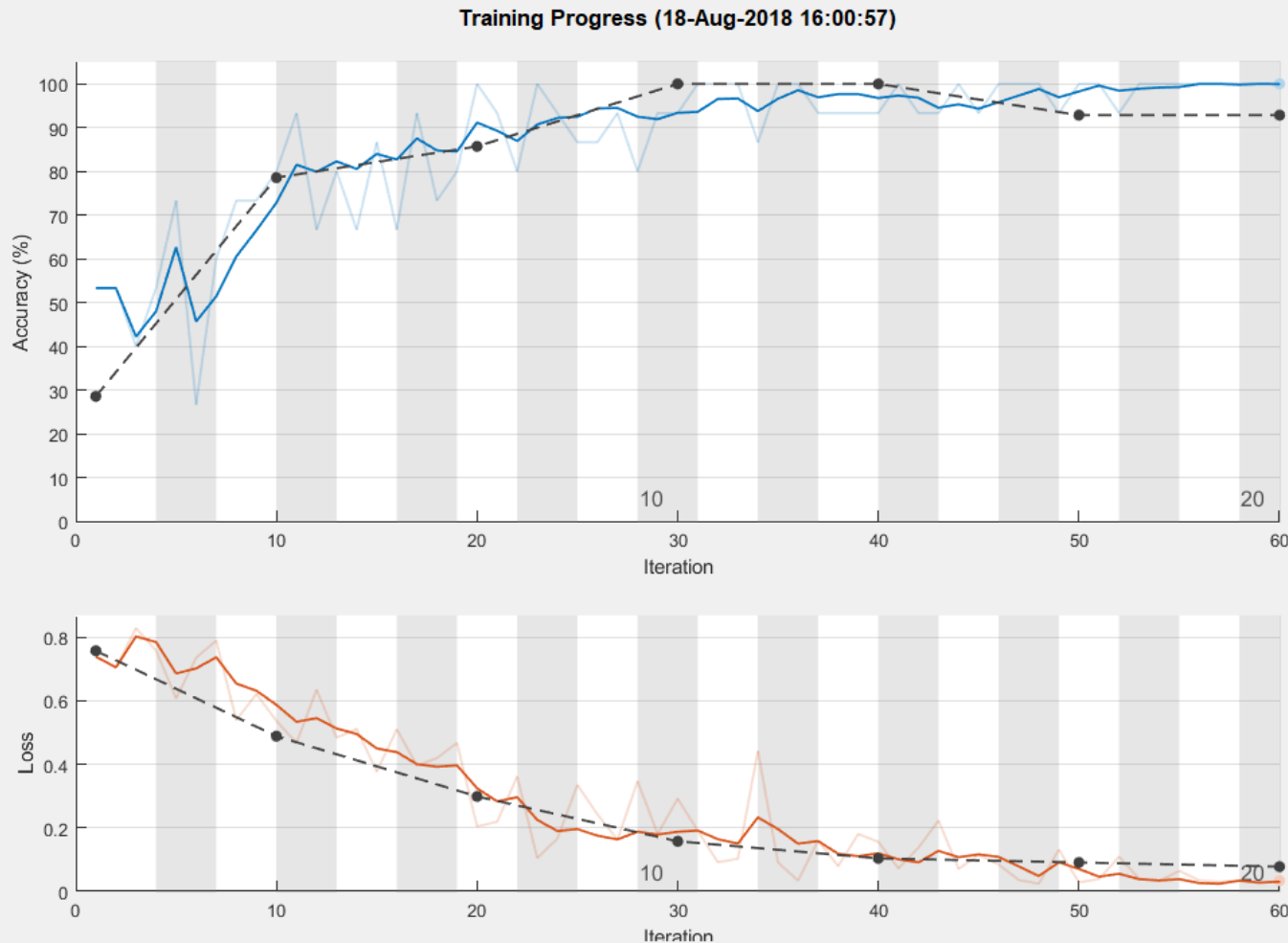
Command Window

Type here to search

VIBRATION ANALYSIS:



Training Progress (18-Aug-2018 16:00:57)



Results

Validation accuracy: 92.86%
Training finished: Reached final iteration

Training Time

Start time: 18-Aug-2018 16:00:57
Elapsed time: 104 min 51 sec

Training Cycle

Epoch: 20 of 20
Iteration: 60 of 60
Iterations per epoch: 3
Maximum iterations: 60

Validation

Frequency: 10 iterations
Patience: Inf

Other Information

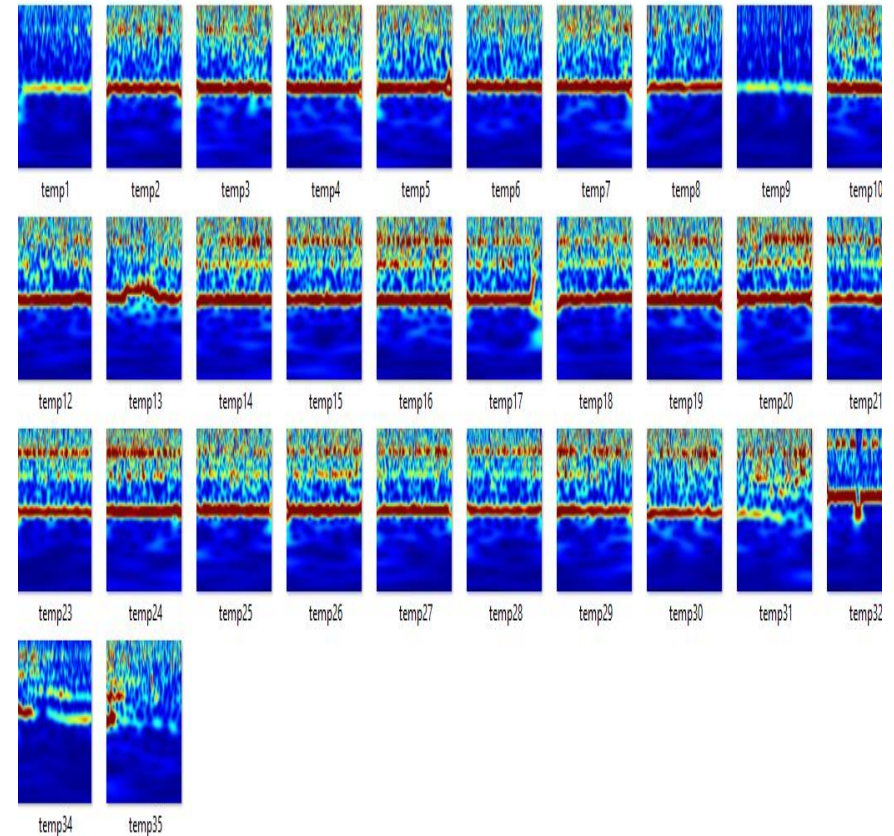
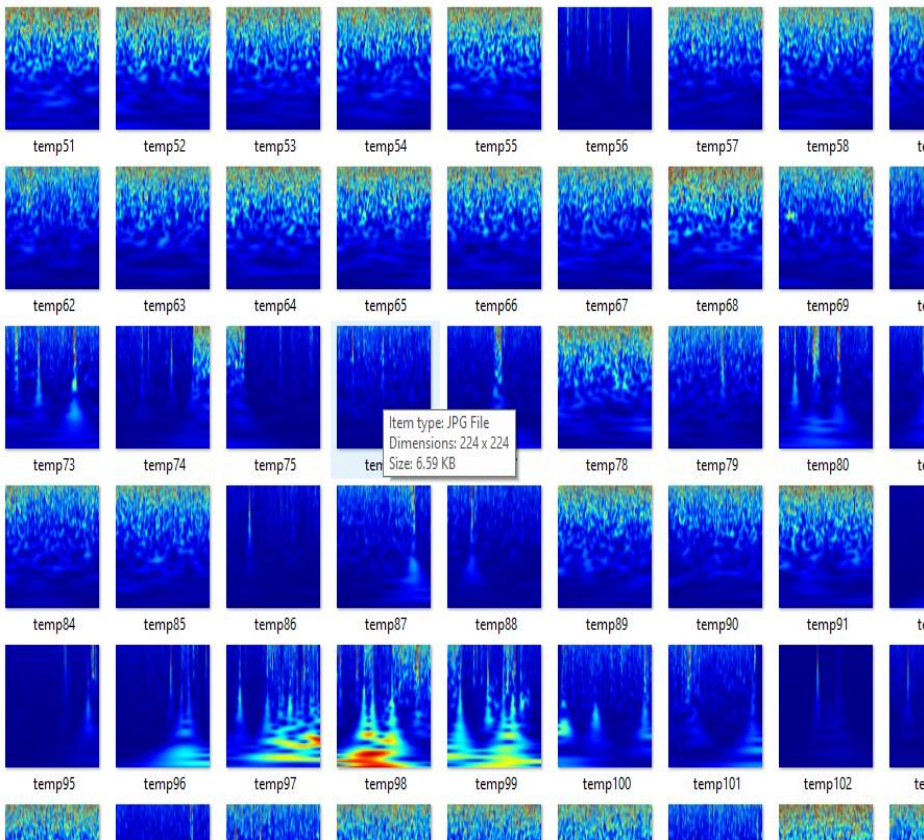
Hardware resource: Single CPU
Learning rate schedule: Constant
Learning rate: 0.0001

Learn more





DATA SET





- **First few layers of the CNN**
 - Input Layer $224*224*3$
 - conv1-7x7_s2 $64 \quad 7*7*3$
 - conv1-relu_7x7 ReLu
 - pool1-3x3_s2 $3*3$ max pooling
- **Last four layers of the CNN**
 - Dropout layer
 - Fully connected layer
 - Softmax layer
 - Class output layer
 - [Vibration.pptx](#) (detailed explanation of ML)



BUSINESS PLAN

Why Smart Slurry Pumping System?



- Slurry is one of the most challenging materials for a pump to handle.
- This is due to uncertainty in the nature of suspended matter.
- Hence the motor would be subjected to fluctuating loading conditions.



Customer Requirement Identification



- Survey is taken from our local industries.
- Continuous monitoring of vibration is required.
- Continuous monitoring of insulation temperature is required.
- Monitoring of standard electrical parameters such as voltage, current, frequency and harmonic distortions are also required
- Retrofitting is most important





Different types of fault in rotating machines in percentage

Causes of different faults

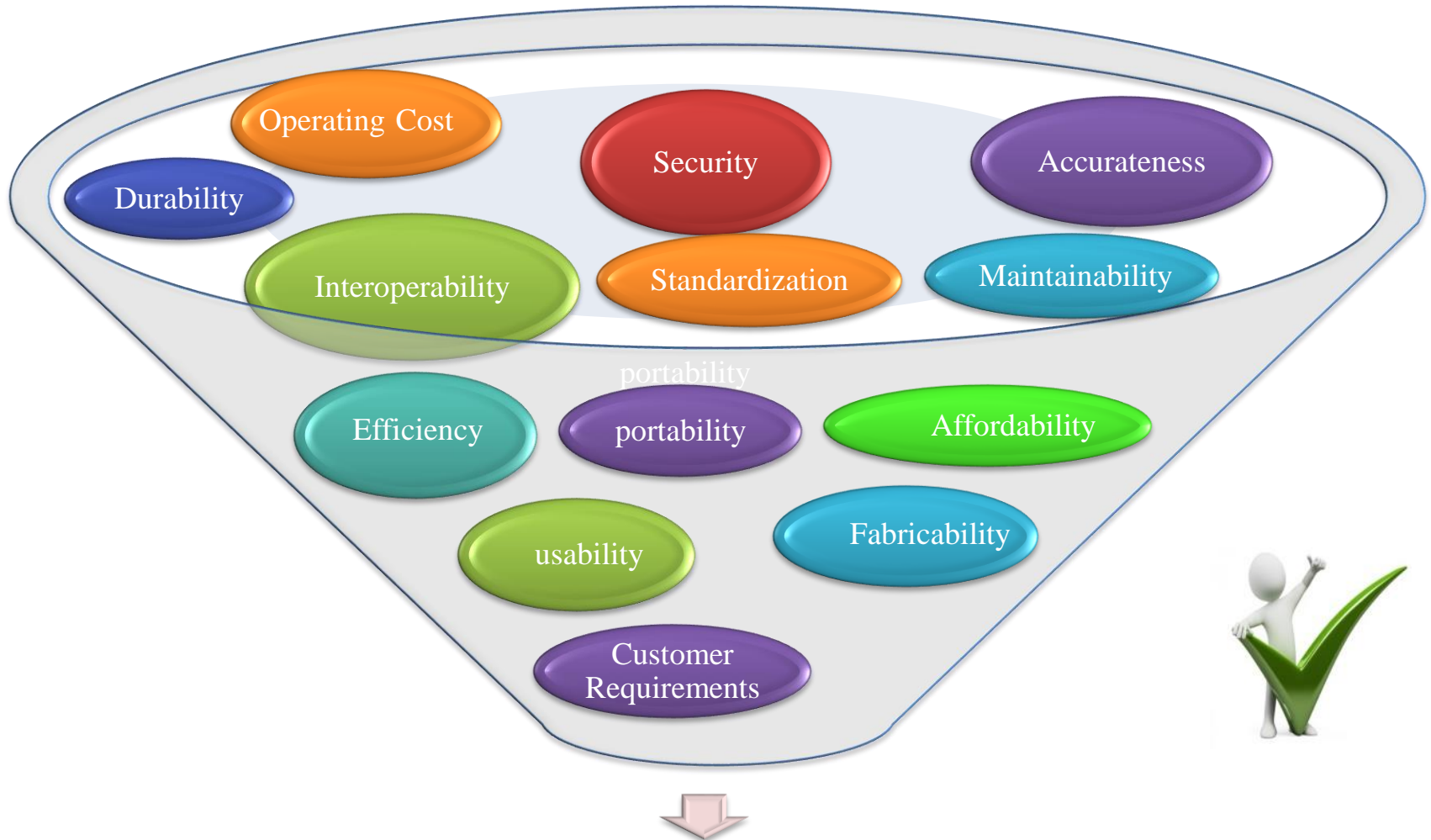
Reason routinely highlighted for all faults are:

Thermal stress – Temperature rise

Mechanical stress – Vibration

Electrical stress – Faults and harmonics

Benefits Of The System



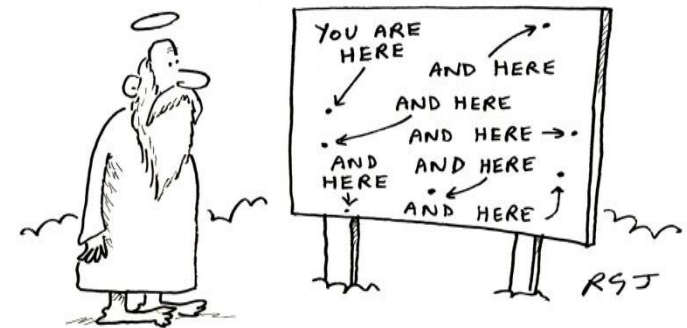
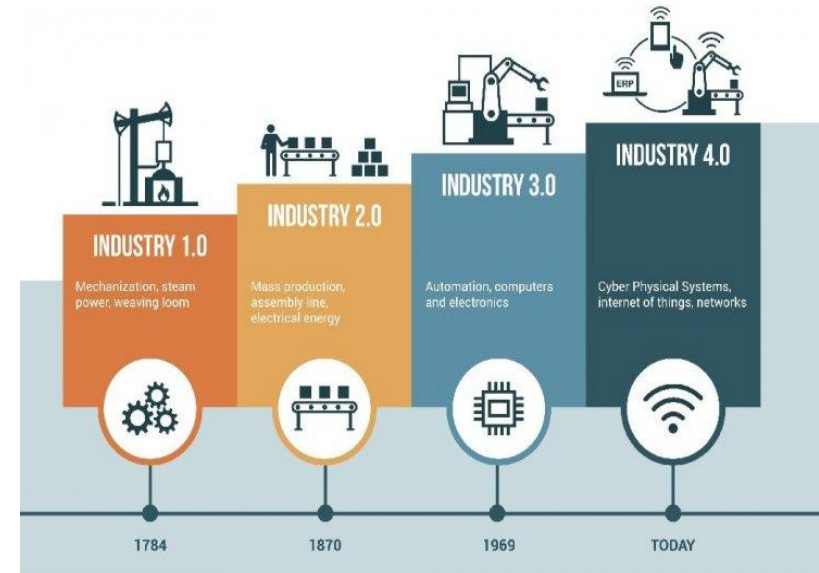
Smart Slurry Pumping System

EngiNx 2018

Uniqueness of the system



- **Industry 4.0**
- **Centralized operation**
- **Less maintenance**
- **Digital Twin**
- **Cloud computing technology**
- **Parallel processing of all parameters.**
- **Generalized Solution to problem of industrial omnipresent i.e. motor.**

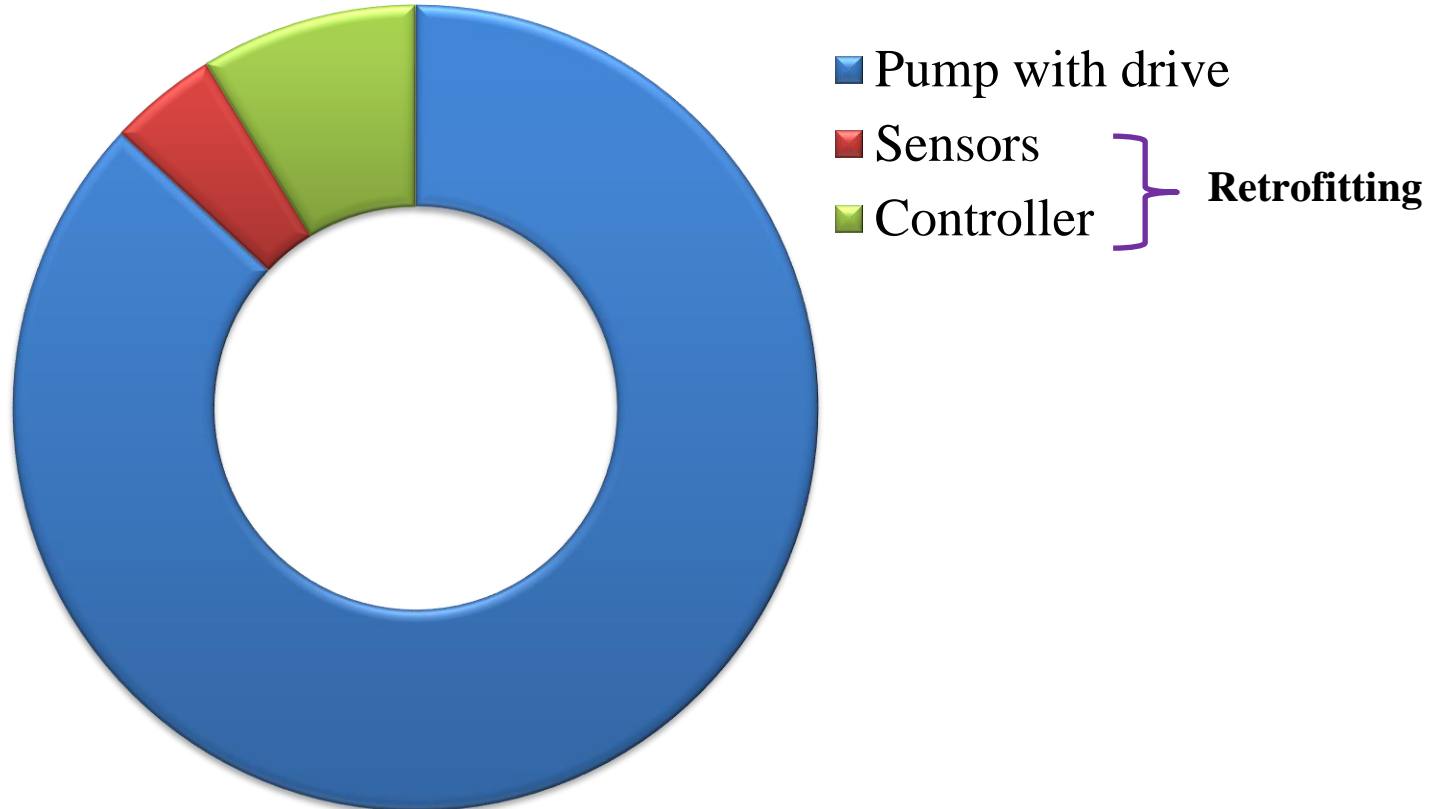


Bill of Materials



S. No.	Components	Amount (Rs.)
1	Tri-axial accelerometer	500
2	Non-contact type temperature sensor	500
3	Contact type temperature sensor	200
4	Raspberry Pi 3 Model B	3000
5	Energy management unit	10000
6	Software development	20000
	TOTAL	34200

Implementation Cost Distribution



Improving Reliability and Uptime



**The Digital Twin
can act before
product failure
begins.**



**Better product
reliability,
increased uptime
and lower
maintenance cost
creates customer
value.**



- **Tease**

What is the problem?

- **Please**

What is the solution?

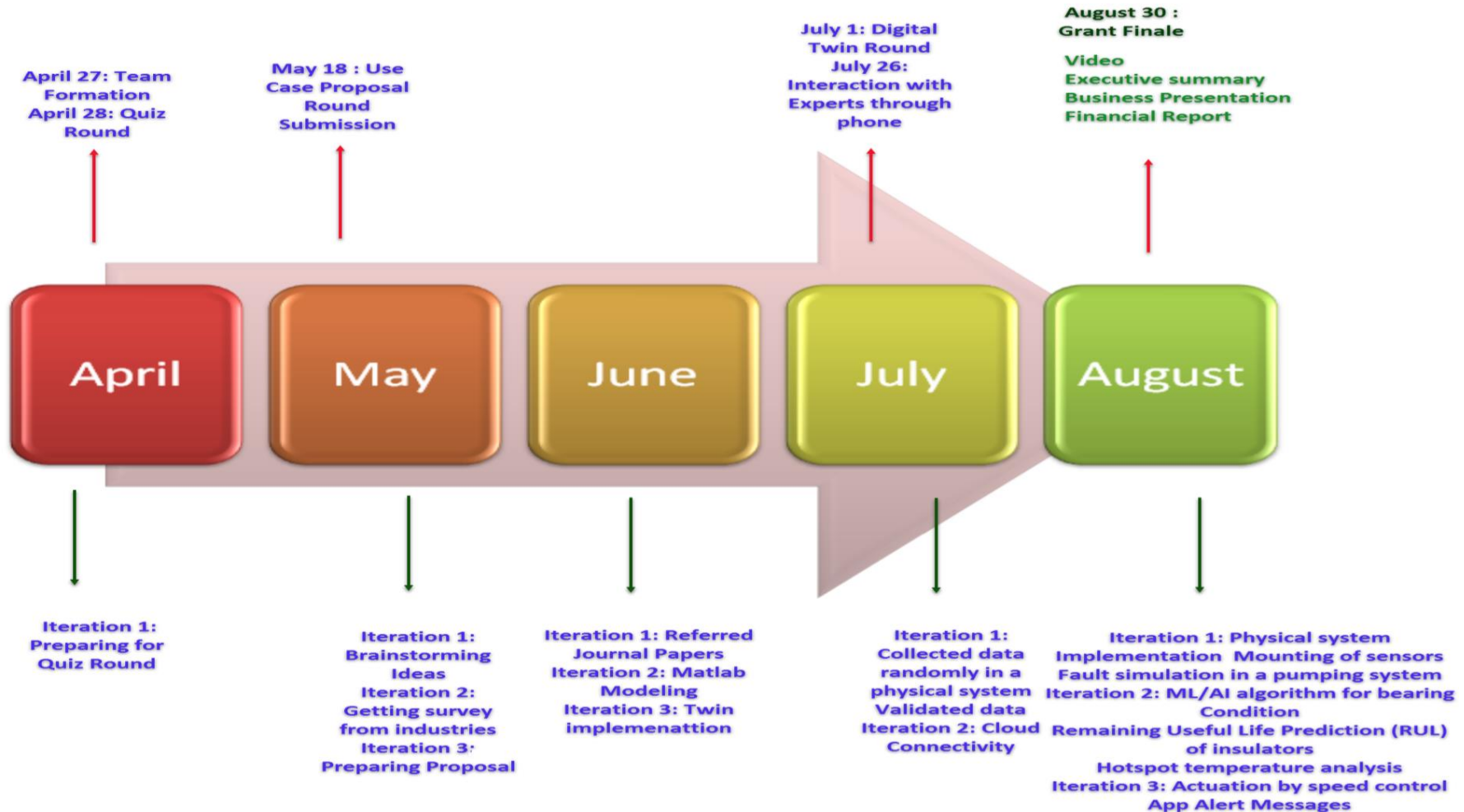
- **Seize**

**What are the
opportunities?**

**Demonstrating
Prototype.**



Milestones





- **Ma, J., Wu, J., & Wang, X. (2018). Fault diagnosis method based on wavelet packet-energy entropy and fuzzy kernel extreme learning machine. Advances in Mechanical Engineering, 10(1), 1687814017751446.**
- **Mathew, J., & Alfredson, R. J. (1984). The condition monitoring of rolling element bearings using vibration analysis. Journal of vibration, acoustics, stress, and reliability in design, 106(3), 447-453.**
- **Kothari DP, Nagrath IJ (2010) Electric machines. Tata McGraw Hill Education Pvt. Ltd., New Delhi. ISBN-13 978-0-07-069967-0**
- **Schoen RR, Habetler TG, Kamran F, Bartheld RG (1995) Motor bearing damage detection using stator current monitoring. IEEE Trans Ind Appl 31(6):1274–1279**
- **Tavner PJ (2008) Review of condition monitoring of rotating electrical machines. IET Electr. Power Appl 2(4), 215**
- **Omar Badran, Hussain Sarhan, Bilal Alomour. Thermal performance analysis of induction motors. International Information and Engineering Technology Association.**



- Bing, L., Yilong, L., & Wei, C. (2018). A fuzzy support vector machine based on environmental membership and its application to motor fault classification. *Journal of Vibration and Control*, 1077546318764484.
- Merainani, B., Rahmoune, C., Benazzouz, D., & Ould-Bouamama, B. (2017). A novel gearbox fault feature extraction and classification using Hilbert empirical wavelet transform, singular value decomposition, and SOM neural network. *Journal of Vibration and Control*, 1077546316688991.
- Rodriguez-Donate, C., Romero-Troncoso, R. J., Cabal-Yepez, E., Garcia-Perez, A., & Osornio-Rios, R. A. (2011). Wavelet-based general methodology for multiple fault detection on induction motors at the startup vibration transient. *Journal of Vibration and Control*, 17(9), 1299-1309.
- Li, W., & Mechefske, C. K. (2006). Detection of induction motor faults: a comparison of stator current, vibration and acoustic methods. *Journal of vibration and Control*, 12(2), 165-188.
- Jayakumar, K., & Thangavel, S. (2017). Industrial drive fault diagnosis through vibration analysis using wavelet transform. *Journal of Vibration and Control*, 23(12), 2003-2013.



THANK YOU