APPENDIX I:
AERODYNAMIC PRESSURE AND HOTWIRE DATA

PRESSURE DATA

The data below shows the compiled data from several tests in the wind tunnel with the pressure transducers, the data included is for completeness since the important conclusions and findings have already been discussed. One interesting thing to note here however, is that, as the model increases in sides, the pressure readings start to resemble the data taken from an experiment on a circular cylinder as should be expected.

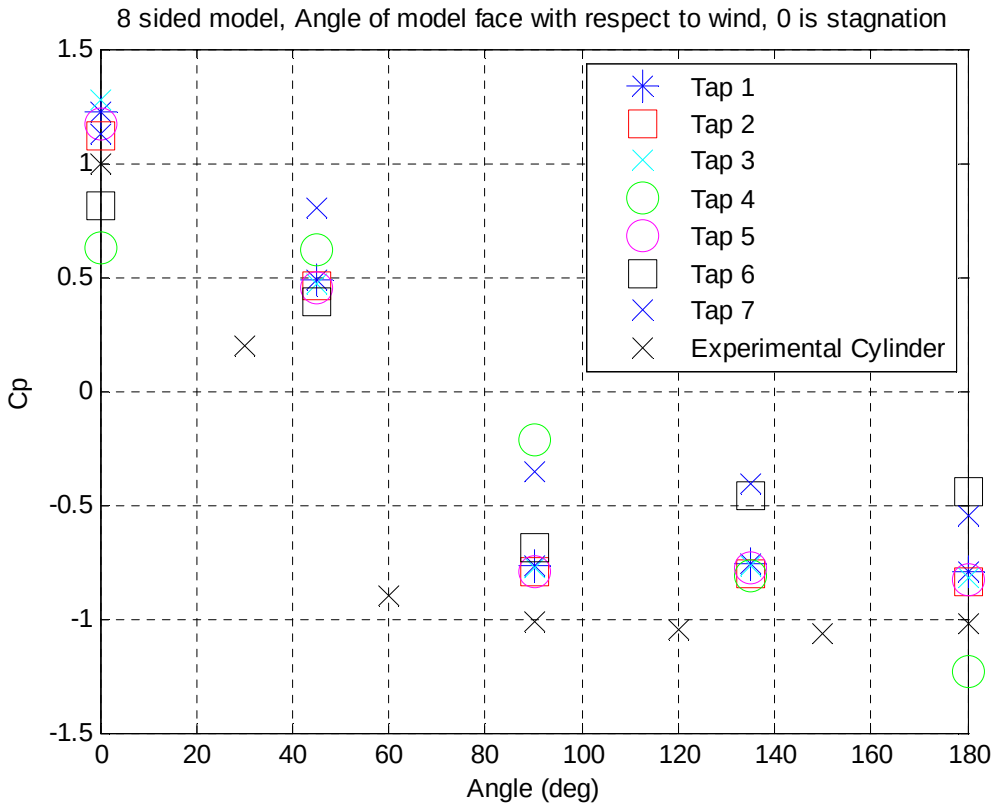

Figure I-1: 8-sided model, pressure readings

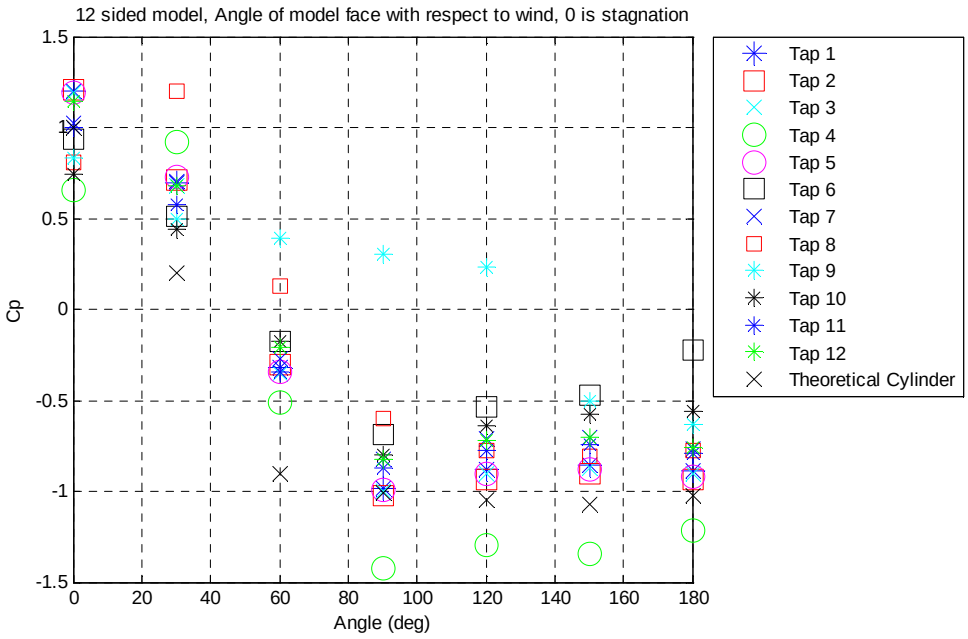

Figure I-2: 12-sided model pressure readings

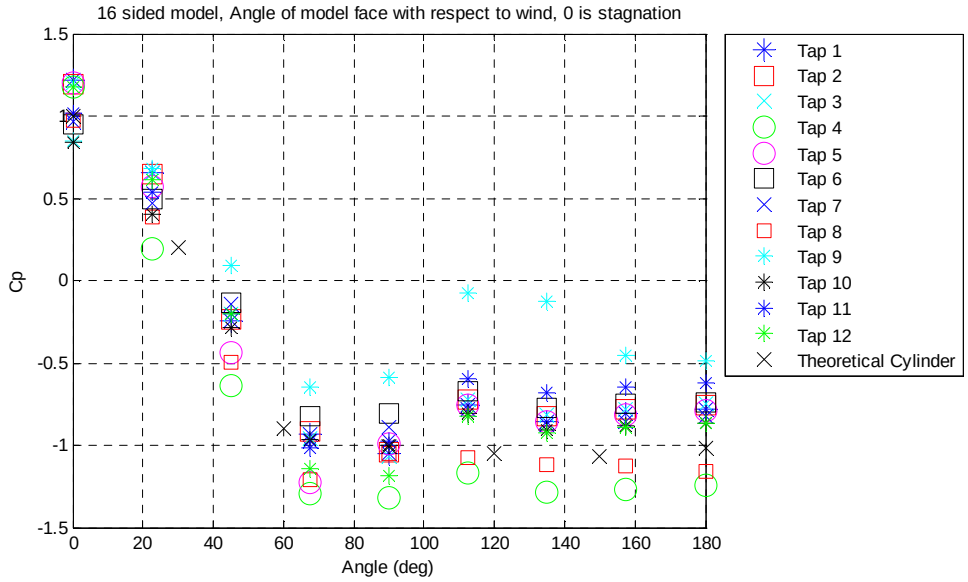

Figure I-3: 16-sided model pressure readings

HOTWIRE DATA

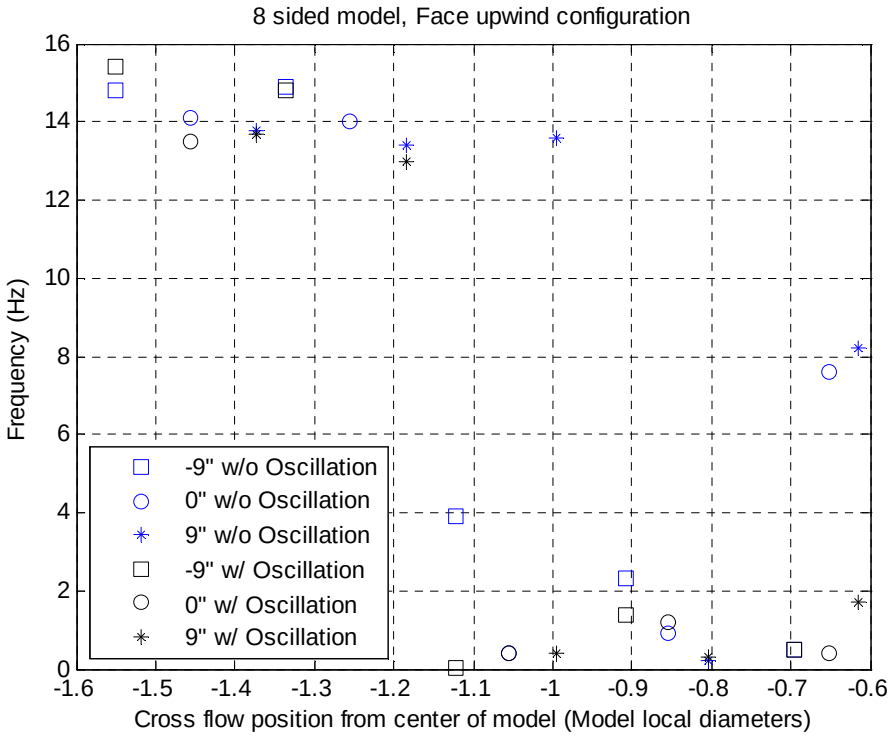

Figure I-4

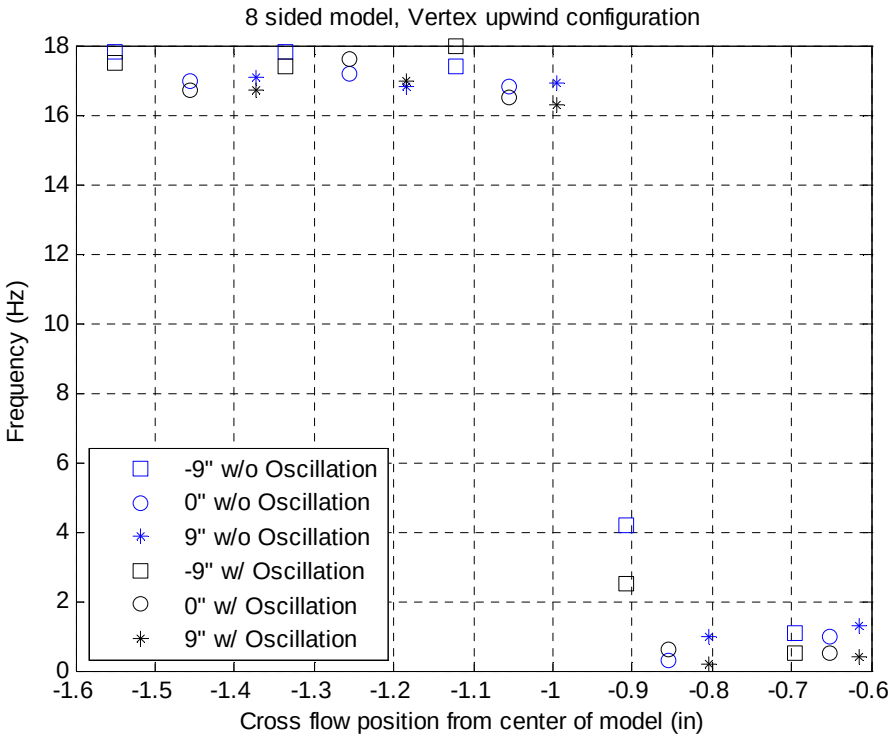

Figure I-5

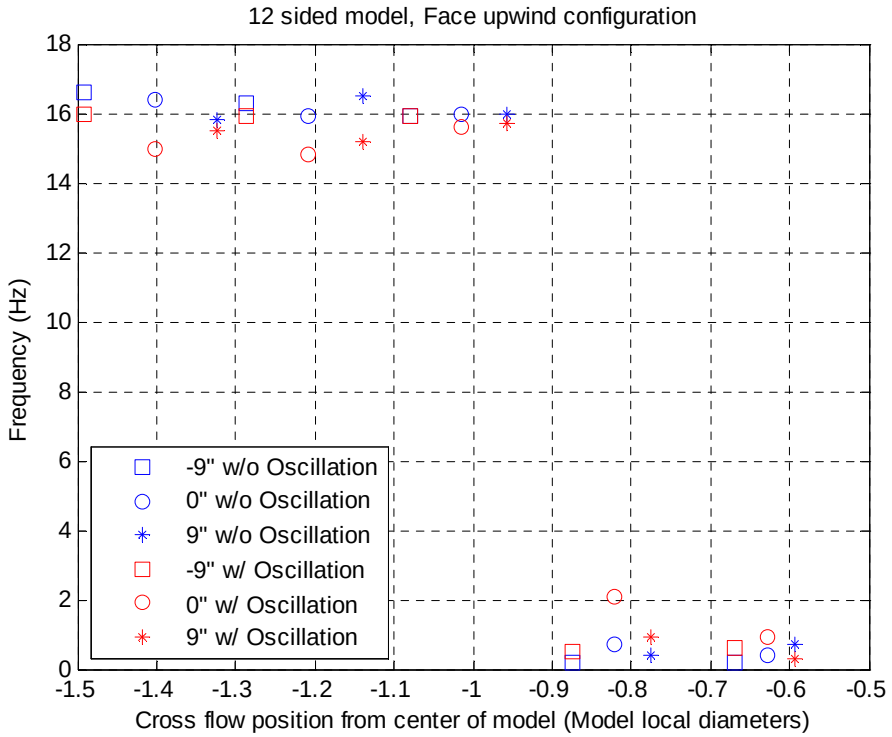

Figure I-6

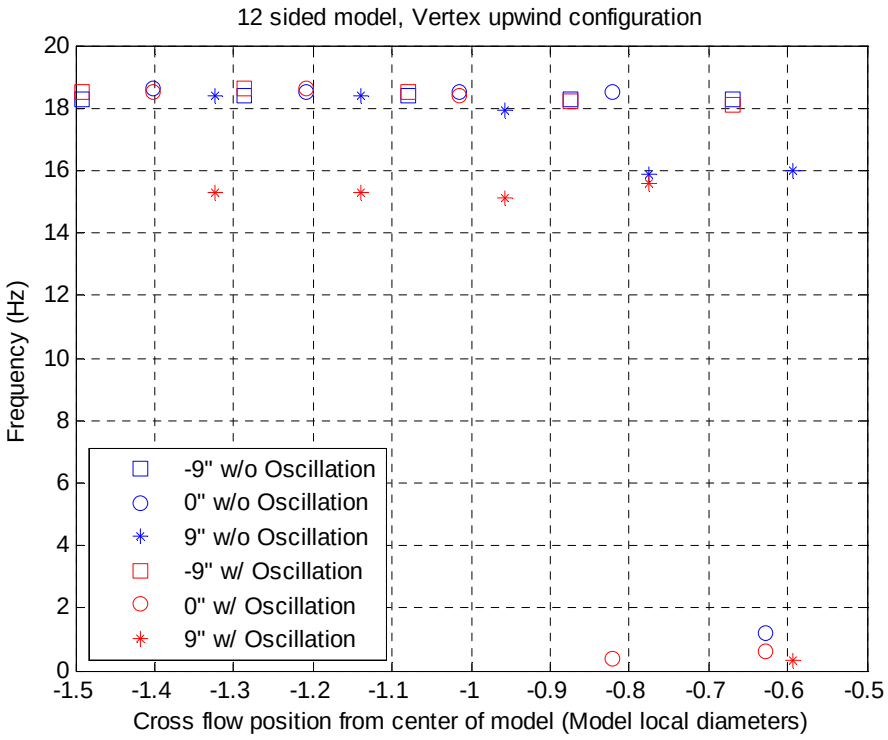

Figure I-7

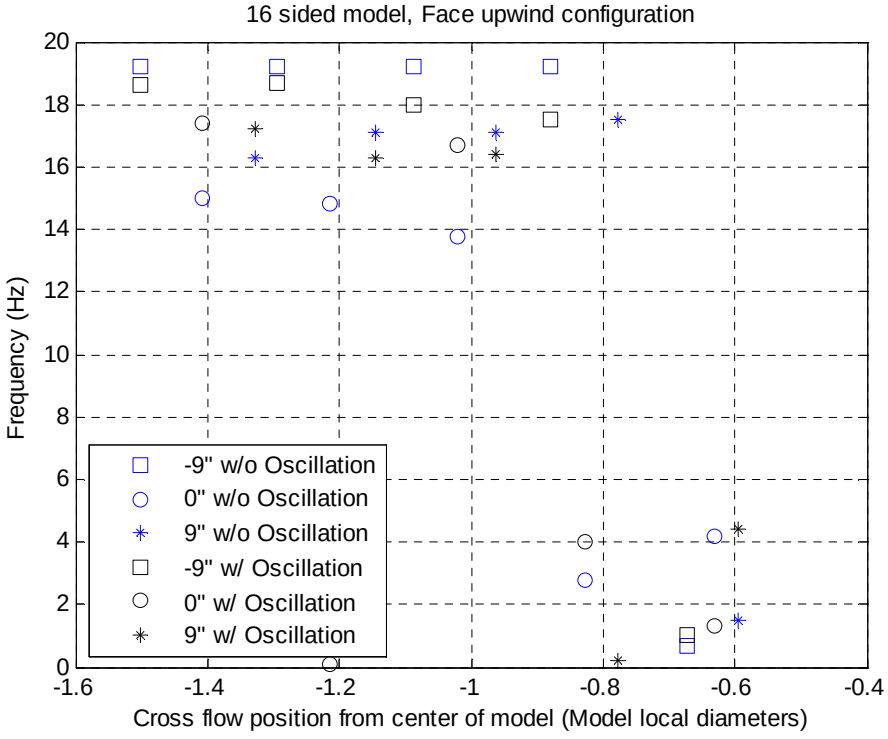

Figure I-8

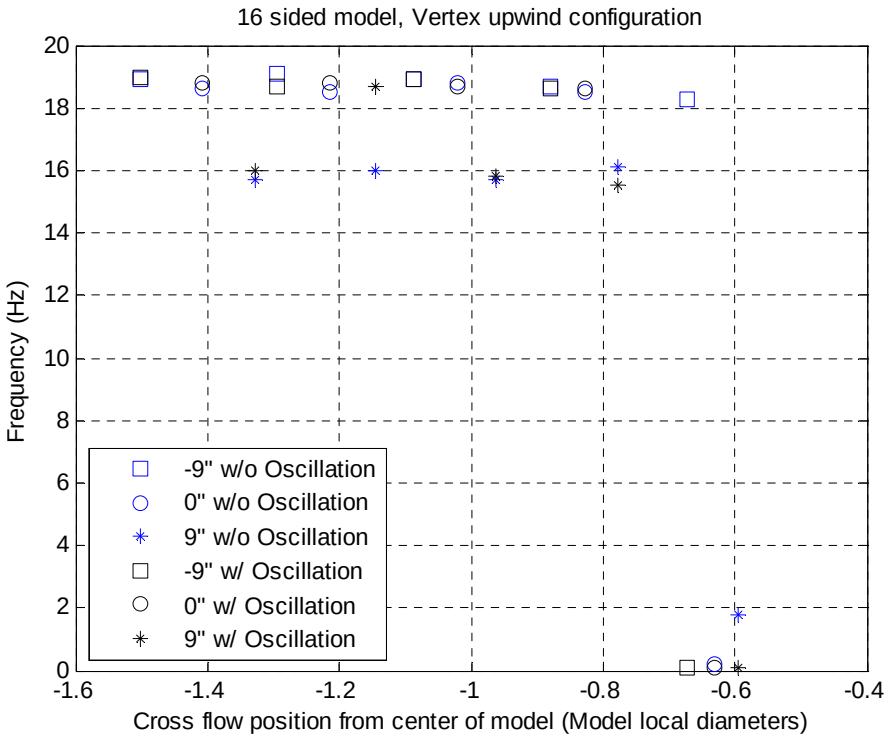

Figure I-9

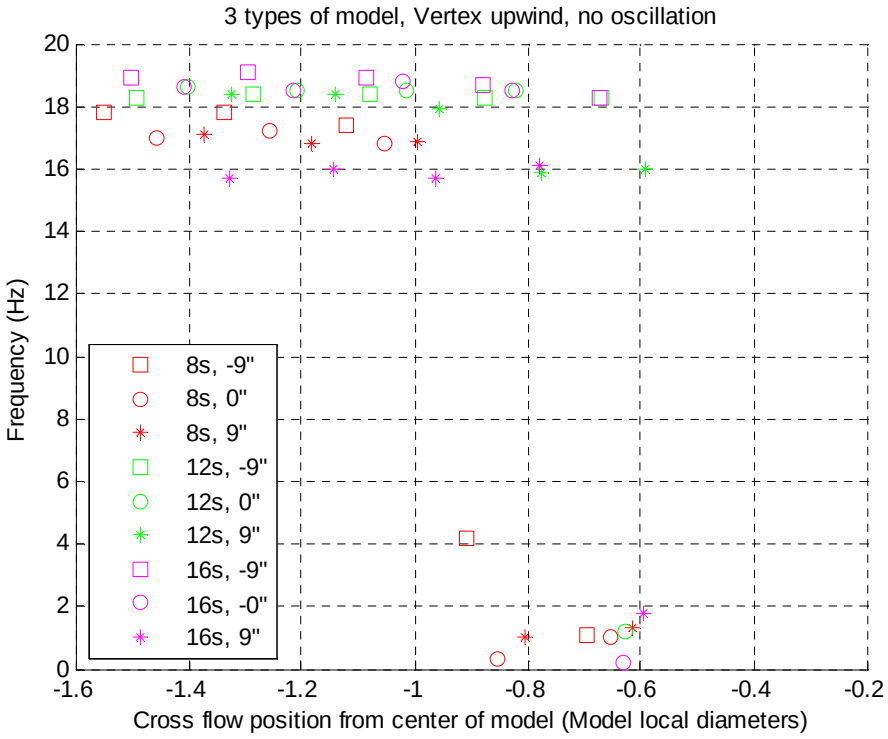

Figure I-10

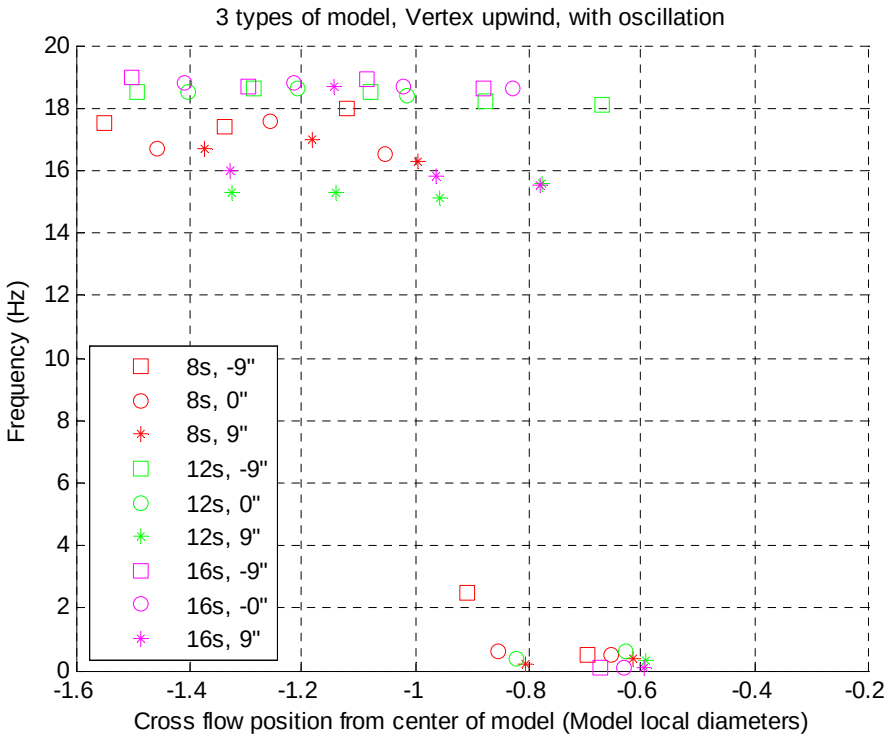
3 types of model, Vertex upwind, with oscillation
Frequency (Hz)
Cross flow position from center of model (Model local diameters)
8s, -9"
8s, 0"
8s, 9"
12s, -9"
12s, 0"
12s, 9"
16s, -9"
16s, -0"
16s, 9"

Figure I-11

AERODYNAMIC CODE

```
Hotwire Processing Program

%% Hotwire data

% Clear
close all
clear all
fclose('all')
clc

%% Distances
d0 = 21.25;
d = d0 - [18,17,16,15,14];

%% Radii at faces and vertex
d8 = [15.5,16.5,17.5];
d12 = [15+5/8,16+5/8,17+5/8];
d16 = [15+3/8,16+3/8,17+3/8];

St = 0.21; % Circular cylinder Strouhal number
V = 9.79; % Flow velocity +- 0.05 m/s
V_en = 9.79/0.0254;

s8 = d8/8;
s12 = d12/12;
s16 = d16/16;

rf8 = s8*0.5/tan(pi/8);
rv8 = s8*0.5/sin(pi/8);
rf12 = s12*0.5/tan(pi/12);
rv12 = s12*0.5/sin(pi/12);
rf16 = s16*0.5/tan(pi/16);
rv16 = s16*0.5/sin(pi/16);

ff8 = St*V./(2*rf8);
fv8 = St*V./(2*rv8);
ff12 = St*V./(2*rf12);
fv12 = St*V./(2*rv12);
ff16 = St*V./(2*rf16);
fv16 = St*V./(2*rv16);

%% Load data
% Stiff springs
% 16 sided Face configuration
[v16sFA(1,:),t16sFA(1,:)] = tekread('16sF18inAp_Ch1.wfm');
[v16sFA(2,:),t16sFA(2,:)] = tekread('16sF17inAp_Ch1.wfm');
[v16sFA(3,:),t16sFA(3,:)] = tekread('16sF16inAp_Ch1.wfm');
[v16sFA(4,:),t16sFA(4,:)] = tekread('16sF15inAp_Ch1.wfm');
[v16sFA(5,:),t16sFA(5,:)] = tekread('16sF14inAp_Ch1.wfm');

[v16sFB(1,:),t16sFB(1,:)] = tekread('16sF18inBp_Ch1.wfm');
[v16sFB(2,:),t16sFB(2,:)] = tekread('16sF17inBp_Ch1.wfm');
```

```
[v16sFB(3,:),t16sFB(3,:)] = tekread('16sF16inBp_Ch1.wfm');
[v16sFB(4,:),t16sFB(4,:)] = tekread('16sF15inBp_Ch1.wfm');
[v16sFB(5,:),t16sFB(5,:)] = tekread('16sF14inBp_Ch1.wfm');

[v16sFC(1,:),t16sFC(1,:)] = tekread('16sF18inCp_Ch1.wfm');
[v16sFC(2,:),t16sFC(2,:)] = tekread('16sF17inCp_Ch1.wfm');
[v16sFC(3,:),t16sFC(3,:)] = tekread('16sF16inCp_Ch1.wfm');
[v16sFC(4,:),t16sFC(4,:)] = tekread('16sF15inCp_Ch1.wfm');
[v16sFC(5,:),t16sFC(5,:)] = tekread('16sF14inCp_Ch1.wfm');
% 16 sided Vertex configuration
[v16sVA(1,:),t16sVA(1,:)] = tekread('16sV18inAp_Ch1.wfm');
[v16sVA(2,:),t16sVA(2,:)] = tekread('16sV17inAp_Ch1.wfm');
[v16sVA(3,:),t16sVA(3,:)] = tekread('16sV16inAp_Ch1.wfm');
[v16sVA(4,:),t16sVA(4,:)] = tekread('16sV15inAp_Ch1.wfm');
[v16sVA(5,:),t16sVA(5,:)] = tekread('16sV14inAp_Ch1.wfm');

[v16sVB(1,:),t16sVB(1,:)] = tekread('16sV18inBp_Ch1.wfm');
[v16sVB(2,:),t16sVB(2,:)] = tekread('16sV17inBp_Ch1.wfm');
[v16sVB(3,:),t16sVB(3,:)] = tekread('16sV16inBp_Ch1.wfm');
[v16sVB(4,:),t16sVB(4,:)] = tekread('16sV15inBp_Ch1.wfm');
[v16sVB(5,:),t16sVB(5,:)] = tekread('16sV14inBp_Ch1.wfm');

[v16sVC(1,:),t16sVC(1,:)] = tekread('16sV18inCp_Ch1.wfm');
[v16sVC(2,:),t16sVC(2,:)] = tekread('16sV17inCp_Ch1.wfm');
[v16sVC(3,:),t16sVC(3,:)] = tekread('16sV16inCp_Ch1.wfm');
[v16sVC(4,:),t16sVC(4,:)] = tekread('16sV15inCp_Ch1.wfm');
[v16sVC(5,:),t16sVC(5,:)] = tekread('16sV14inCp_Ch1.wfm');
% 12 sided Face configuration
[v12sFA(1,:),t12sFA(1,:)] = tekread('12sF18inAp_Ch1.wfm');
[v12sFA(2,:),t12sFA(2,:)] = tekread('12sF17inAp_Ch1.wfm');
[v12sFA(3,:),t12sFA(3,:)] = tekread('12sF16inAp_Ch1.wfm');
[v12sFA(4,:),t12sFA(4,:)] = tekread('12sF15inAp_Ch1.wfm');
[v12sFA(5,:),t12sFA(5,:)] = tekread('12sF14inAp_Ch1.wfm');

[v12sFB(1,:),t12sFB(1,:)] = tekread('12sF18inBp_Ch1.wfm');
[v12sFB(2,:),t12sFB(2,:)] = tekread('12sF17inBp_Ch1.wfm');
[v12sFB(3,:),t12sFB(3,:)] = tekread('12sF16inBp_Ch1.wfm');
[v12sFB(4,:),t12sFB(4,:)] = tekread('12sF15inBp_Ch1.wfm');
[v12sFB(5,:),t12sFB(5,:)] = tekread('12sF14inBp_Ch1.wfm');

[v12sFC(1,:),t12sFC(1,:)] = tekread('12sF18inCp_Ch1.wfm');
[v12sFC(2,:),t12sFC(2,:)] = tekread('12sF17inCp_Ch1.wfm');
[v12sFC(3,:),t12sFC(3,:)] = tekread('12sF16inCp_Ch1.wfm');
[v12sFC(4,:),t12sFC(4,:)] = tekread('12sF15inCp_Ch1.wfm');
[v12sFC(5,:),t12sFC(5,:)] = tekread('12sF14inCp_Ch1.wfm');
% 12 sided Vertex configuration
[v12sVA(1,:),t12sVA(1,:)] = tekread('12sV18inAp_Ch1.wfm');
[v12sVA(2,:),t12sVA(2,:)] = tekread('12sV17inAp_Ch1.wfm');
[v12sVA(3,:),t12sVA(3,:)] = tekread('12sV16inAp_Ch1.wfm');
[v12sVA(4,:),t12sVA(4,:)] = tekread('12sV15inAp_Ch1.wfm');
[v12sVA(5,:),t12sVA(5,:)] = tekread('12sV14inAp_Ch1.wfm');

[v12sVB(1,:),t12sVB(1,:)] = tekread('12sV18inBp_Ch1.wfm');
[v12sVB(2,:),t12sVB(2,:)] = tekread('12sV17inBp_Ch1.wfm');
[v12sVB(3,:),t12sVB(3,:)] = tekread('12sV16inBp_Ch1.wfm');
[v12sVB(4,:),t12sVB(4,:)] = tekread('12sV15inBp_Ch1.wfm');
```

```
[v12sVB(5,:),t12sVB(5,:)] = tekread('12sV14inBp_Ch1.wfm');

[v12sVC(1,:),t12sVC(1,:)] = tekread('12sV18inCp_Ch1.wfm');
[v12sVC(2,:),t12sVC(2,:)] = tekread('12sV17inCp_Ch1.wfm');
[v12sVC(3,:),t12sVC(3,:)] = tekread('12sV16inCp_Ch1.wfm');
[v12sVC(4,:),t12sVC(4,:)] = tekread('12sV15inCp_Ch1.wfm');
[v12sVC(5,:),t12sVC(5,:)] = tekread('12sV14inCp_Ch1.wfm');
% 8 sided Face configuration
[v8sFA(1,:),t8sFA(1,:)] = tekread('8sF18inAp_Ch1.wfm');
[v8sFA(2,:),t8sFA(2,:)] = tekread('8sF17inAp_Ch1.wfm');
[v8sFA(3,:),t8sFA(3,:)] = tekread('8sF16inAp_Ch1.wfm');
[v8sFA(4,:),t8sFA(4,:)] = tekread('8sF15inAp_Ch1.wfm');
[v8sFA(5,:),t8sFA(5,:)] = tekread('8sF14inAp_Ch1.wfm');

[v8sFB(1,:),t8sFB(1,:)] = tekread('8sF18inBp_Ch1.wfm');
[v8sFB(2,:),t8sFB(2,:)] = tekread('8sF17inBp_Ch1.wfm');
[v8sFB(3,:),t8sFB(3,:)] = tekread('8sF16inBp_Ch1.wfm');
[v8sFB(4,:),t8sFB(4,:)] = tekread('8sF15inBp_Ch1.wfm');
[v8sFB(5,:),t8sFB(5,:)] = tekread('8sF14inBp_Ch1.wfm');

[v8sFC(1,:),t8sFC(1,:)] = tekread('8sF18inCp_Ch1.wfm');
[v8sFC(2,:),t8sFC(2,:)] = tekread('8sF17inCp_Ch1.wfm');
[v8sFC(3,:),t8sFC(3,:)] = tekread('8sF16inCp_Ch1.wfm');
[v8sFC(4,:),t8sFC(4,:)] = tekread('8sF15inCp_Ch1.wfm');
[v8sFC(5,:),t8sFC(5,:)] = tekread('8sF14inCp_Ch1.wfm');
% 8 sided Vertex configuration
[v8sVA(1,:),t8sVA(1,:)] = tekread('8sV18inAp_Ch1.wfm');
[v8sVA(2,:),t8sVA(2,:)] = tekread('8sV17inAp_Ch1.wfm');
[v8sVA(3,:),t8sVA(3,:)] = tekread('8sV16inAp_Ch1.wfm');
[v8sVA(4,:),t8sVA(4,:)] = tekread('8sV15inAp_Ch1.wfm');
[v8sVA(5,:),t8sVA(5,:)] = tekread('8sV14inAp_Ch1.wfm');

[v8sVB(1,:),t8sVB(1,:)] = tekread('8sV18inBp_Ch1.wfm');
[v8sVB(2,:),t8sVB(2,:)] = tekread('8sV17inBp_Ch1.wfm');
[v8sVB(3,:),t8sVB(3,:)] = tekread('8sV16inBp_Ch1.wfm');
[v8sVB(4,:),t8sVB(4,:)] = tekread('8sV15inBp_Ch1.wfm');
[v8sVB(5,:),t8sVB(5,:)] = tekread('8sV14inBp_Ch1.wfm');

[v8sVC(1,:),t8sVC(1,:)] = tekread('8sV18inCp_Ch1.wfm');
[v8sVC(2,:),t8sVC(2,:)] = tekread('8sV17inCp_Ch1.wfm');
[v8sVC(3,:),t8sVC(3,:)] = tekread('8sV16inCp_Ch1.wfm');
[v8sVC(4,:),t8sVC(4,:)] = tekread('8sV15inCp_Ch1.wfm');
[v8sVC(5,:),t8sVC(5,:)] = tekread('8sV14inCp_Ch1.wfm');

% With the Less stiff springs
% 16 sided Face configuration
[v16sFAO(1,:),t16sFAO(1,:)] = tekread('16sF18inApO.wfm');
[v16sFAO(2,:),t16sFAO(2,:)] = tekread('16sF17inApO.wfm');
[v16sFAO(3,:),t16sFAO(3,:)] = tekread('16sF16inApO.wfm');
[v16sFAO(4,:),t16sFAO(4,:)] = tekread('16sF15inApO.wfm');
[v16sFAO(5,:),t16sFAO(5,:)] = tekread('16sF14inApO.wfm');

[v16sFBO(1,:),t16sFBO(1,:)] = tekread('16sF18inBpO.wfm');
[v16sFBO(2,:),t16sFBO(2,:)] = tekread('16sF17inBpO.wfm');
[v16sFBO(3,:),t16sFBO(3,:)] = tekread('16sF16inBpO.wfm');
```

```
[v16sFBO(4,:),t16sFBO(4,:)] = tekread('16sF15inBpO.wfm');
[v16sFBO(5,:),t16sFBO(5,:)] = tekread('16sF14inBpO.wfm');

[v16sFCO(1,:),t16sFCO(1,:)] = tekread('16sF18inCpO.wfm');
[v16sFCO(2,:),t16sFCO(2,:)] = tekread('16sF17inCpO.wfm');
[v16sFCO(3,:),t16sFCO(3,:)] = tekread('16sF16inCpO.wfm');
[v16sFCO(4,:),t16sFCO(4,:)] = tekread('16sF15inCpO.wfm');
[v16sFCO(5,:),t16sFCO(5,:)] = tekread('16sF14inCpO.wfm');
% 16 sided Vertex configuration
% [v16sVAO(1,:),t16sVAO(1,:)] = tekread('16sV18inApO.wfm');
[v16sVAO(2,:),t16sVAO(2,:)] = tekread('16sV17inApO.wfm');
[v16sVAO(3,:),t16sVAO(3,:)] = tekread('16sV16inApO.wfm');
[v16sVAO(4,:),t16sVAO(4,:)] = tekread('16sV15inApO.wfm');
[v16sVAO(5,:),t16sVAO(5,:)] = tekread('16sV14inApO.wfm');

% [v16sVBO(1,:),t16sVBO(1,:)] = tekread('16sV18inBpO.wfm');
[v16sVBO(2,:),t16sVBO(2,:)] = tekread('16sV17inBpO.wfm');
[v16sVBO(3,:),t16sVBO(3,:)] = tekread('16sV16inBpO.wfm');
[v16sVBO(4,:),t16sVBO(4,:)] = tekread('16sV15inBpO.wfm');
[v16sVBO(5,:),t16sVBO(5,:)] = tekread('16sV14inBpO.wfm');

% [v16sVCO(1,:),t16sVCO(1,:)] = tekread('16sV18inCpO.wfm');
[v16sVCO(2,:),t16sVCO(2,:)] = tekread('16sV17inCpO.wfm');
[v16sVCO(3,:),t16sVCO(3,:)] = tekread('16sV16inCpO.wfm');
[v16sVCO(4,:),t16sVCO(4,:)] = tekread('16sV15inCpO.wfm');
[v16sVCO(5,:),t16sVCO(5,:)] = tekread('16sV14inCpO.wfm');
% 12 sided Face configuration
[v12sFAO(1,:),t12sFAO(1,:)] = tekread('12sF18inApO.wfm');
[v12sFAO(2,:),t12sFAO(2,:)] = tekread('12sF17inApO.wfm');
[v12sFAO(3,:),t12sFAO(3,:)] = tekread('12sF16inApO.wfm');
[v12sFAO(4,:),t12sFAO(4,:)] = tekread('12sF15inApO.wfm');
[v12sFAO(5,:),t12sFAO(5,:)] = tekread('12sF14inApO.wfm');

[v12sFBO(1,:),t12sFBO(1,:)] = tekread('12sF18inBpO.wfm');
[v12sFBO(2,:),t12sFBO(2,:)] = tekread('12sF17inBpO.wfm');
[v12sFBO(3,:),t12sFBO(3,:)] = tekread('12sF16inBpO.wfm');
[v12sFBO(4,:),t12sFBO(4,:)] = tekread('12sF15inBpO.wfm');
[v12sFBO(5,:),t12sFBO(5,:)] = tekread('12sF14inBpO.wfm');

[v12sFCO(1,:),t12sFCO(1,:)] = tekread('12sF18inCpO.wfm');
[v12sFCO(2,:),t12sFCO(2,:)] = tekread('12sF17inCpO.wfm');
[v12sFCO(3,:),t12sFCO(3,:)] = tekread('12sF16inCpO.wfm');
[v12sFCO(4,:),t12sFCO(4,:)] = tekread('12sF15inCpO.wfm');
[v12sFCO(5,:),t12sFCO(5,:)] = tekread('12sF14inCpO.wfm');
% 12 sided Vertex configuration
[v12sVAO(1,:),t12sVAO(1,:)] = tekread('12sV18inApO.wfm');
[v12sVAO(2,:),t12sVAO(2,:)] = tekread('12sV17inApO.wfm');
[v12sVAO(3,:),t12sVAO(3,:)] = tekread('12sV16inApO.wfm');
[v12sVAO(4,:),t12sVAO(4,:)] = tekread('12sV15inApO.wfm');
[v12sVAO(5,:),t12sVAO(5,:)] = tekread('12sV14inApO.wfm');

[v12sVBO(1,:),t12sVBO(1,:)] = tekread('12sV18inBpO.wfm');
[v12sVBO(2,:),t12sVBO(2,:)] = tekread('12sV17inBpO.wfm');
[v12sVBO(3,:),t12sVBO(3,:)] = tekread('12sV16inBpO.wfm');
[v12sVBO(4,:),t12sVBO(4,:)] = tekread('12sV15inBpO.wfm');
[v12sVBO(5,:),t12sVBO(5,:)] = tekread('12sV14inBpO.wfm');
```

```
[v12sVCO(1,:),t12sVCO(1,:)] = tekread('12sV18inCpO.wfm');
[v12sVCO(2,:),t12sVCO(2,:)] = tekread('12sV17inCpO.wfm');
[v12sVCO(3,:),t12sVCO(3,:)] = tekread('12sV16inCpO.wfm');
[v12sVCO(4,:),t12sVCO(4,:)] = tekread('12sV15inCpO.wfm');
[v12sVCO(5,:),t12sVCO(5,:)] = tekread('12sV14inCpO.wfm');
% 8 sided Face configuration
[v8sFAO(1,:),t8sFAO(1,:)] = tekread('8sF18inApO.wfm');
[v8sFAO(2,:),t8sFAO(2,:)] = tekread('8sF17inApO.wfm');
[v8sFAO(3,:),t8sFAO(3,:)] = tekread('8sF16inApO.wfm');
[v8sFAO(4,:),t8sFAO(4,:)] = tekread('8sF15inApO.wfm');
[v8sFAO(5,:),t8sFAO(5,:)] = tekread('8sF14inApO.wfm');

[v8sFBO(1,:),t8sFBO(1,:)] = tekread('8sF18inBpO.wfm');
[v8sFBO(2,:),t8sFBO(2,:)] = tekread('8sF17inBpO.wfm');
[v8sFBO(3,:),t8sFBO(3,:)] = tekread('8sF16inBpO.wfm');
[v8sFBO(4,:),t8sFBO(4,:)] = tekread('8sF15inBpO.wfm');
[v8sFBO(5,:),t8sFBO(5,:)] = tekread('8sF14inBpO.wfm');

[v8sFCO(1,:),t8sFCO(1,:)] = tekread('8sF18inCpO.wfm');
[v8sFCO(2,:),t8sFCO(2,:)] = tekread('8sF17inCpO.wfm');
[v8sFCO(3,:),t8sFCO(3,:)] = tekread('8sF16inCpO.wfm');
[v8sFCO(4,:),t8sFCO(4,:)] = tekread('8sF15inCpO.wfm');
[v8sFCO(5,:),t8sFCO(5,:)] = tekread('8sF14inCpO.wfm');
% 8 sided Vertex configuration
[v8sVAO(1,:),t8sVAO(1,:)] = tekread('8sV18inApO.wfm');
[v8sVAO(2,:),t8sVAO(2,:)] = tekread('8sV17inApO.wfm');
[v8sVAO(3,:),t8sVAO(3,:)] = tekread('8sV16inApO.wfm');
[v8sVAO(4,:),t8sVAO(4,:)] = tekread('8sV15inApO.wfm');
[v8sVAO(5,:),t8sVAO(5,:)] = tekread('8sV14inApO.wfm');

[v8sVBO(1,:),t8sVBO(1,:)] = tekread('8sV18inBpO.wfm');
[v8sVBO(2,:),t8sVBO(2,:)] = tekread('8sV17inBpO.wfm');
[v8sVBO(3,:),t8sVBO(3,:)] = tekread('8sV16inBpO.wfm');
[v8sVBO(4,:),t8sVBO(4,:)] = tekread('8sV15inBpO.wfm');
[v8sVBO(5,:),t8sVBO(5,:)] = tekread('8sV14inBpO.wfm');

[v8sVCO(1,:),t8sVCO(1,:)] = tekread('8sV18inCpO.wfm');
[v8sVCO(2,:),t8sVCO(2,:)] = tekread('8sV17inCpO.wfm');
[v8sVCO(3,:),t8sVCO(3,:)] = tekread('8sV16inCpO.wfm');
[v8sVCO(4,:),t8sVCO(4,:)] = tekread('8sV15inCpO.wfm');
[v8sVCO(5,:),t8sVCO(5,:)] = tekread('8sV14inCpO.wfm');


v12sVCh1(1,:) = v12sVC(1,1:500);
v12sVCh1(2,:) = v12sVC(1,501:1000);
v12sVCh1(3,:) = v12sVC(1,1001:1500);
v12sVCh1(4,:) = v12sVC(1,1501:2000);
v12sVCh1(5,:) = v12sVC(1,2001:2500);

v12sVCh2(1,:) = v12sVC(2,1:500);
v12sVCh2(2,:) = v12sVC(2,501:1000);
v12sVCh2(3,:) = v12sVC(2,1001:1500);
v12sVCh2(4,:) = v12sVC(2,1501:2000);
v12sVCh2(5,:) = v12sVC(2,2001:2500);
```

```
v12sVCh3(1,:) = v12sVC(3,1:500);
v12sVCh3(2,:) = v12sVC(3,501:1000);
v12sVCh3(3,:) = v12sVC(3,1001:1500);
v12sVCh3(4,:) = v12sVC(3,1501:2000);
v12sVCh3(5,:) = v12sVC(3,2001:2500);

v12sVCh4(1,:) = v12sVC(4,1:500);
v12sVCh4(2,:) = v12sVC(4,501:1000);
v12sVCh4(3,:) = v12sVC(4,1001:1500);
v12sVCh4(4,:) = v12sVC(4,1501:2000);
v12sVCh4(5,:) = v12sVC(4,2001:2500);

v12sVCh5(1,:) = v12sVC(5,1:500);
v12sVCh5(2,:) = v12sVC(5,501:1000);
v12sVCh5(3,:) = v12sVC(5,1001:1500);
v12sVCh5(4,:) = v12sVC(5,1501:2000);
v12sVCh5(5,:) = v12sVC(5,2001:2500);


%%

%% Start data processing
fs = 250;                                   % Sampling frequency
fs2 = 500;
L = size(v16sFA(1,:),2);                    % Size of data
L2 = size(v16sFB(1,:),2);
Lch = size(v12sVCh1(1,:),2);
f = (1:L/2)/(L/2)*fs;                       % Frequency window
f2 = (1:L2/2)/(L2/2)*fs2;
fch = (1:Lch/2)/(Lch/2)*fs;
w = 0.5*(1-cos(2*pi*(0:(L-1))/(L-1)));      % Window function
w2 = 0.5*(1-cos(2*pi*(0:(L2-1))/(L2-1)));
wch = 0.5*(1-cos(2*pi*(0:(Lch-1))/(Lch-1)));

v12sVB2 = v12sVB;

for i  = 1:5
    % Remove the mean and window it
    v16sFA(i,:) = w.*(v16sFA(i,:)-mean(v16sFA(i,20:L)));
    v16sFB(i,:) = w2.*(v16sFB(i,:)-mean(v16sFB(i,20:L2)));
    v16sFC(i,:) = w.*(v16sFC(i,:)-mean(v16sFC(i,20:L)));
    v16sVA(i,:) = w.*(v16sVA(i,:)-mean(v16sVA(i,20:L)));
    v16sVB(i,:) = w.*(v16sVB(i,:)-mean(v16sVB(i,20:L)));
    v16sVC(i,:) = w.*(v16sVC(i,:)-mean(v16sVC(i,20:L)));

    v12sFA(i,:) = w.*(v12sFA(i,:)-mean(v12sFA(i,20:L)));
    v12sFB(i,:) = w.*(v12sFB(i,:)-mean(v12sFB(i,20:L)));
    v12sFC(i,:) = w.*(v12sFC(i,:)-mean(v12sFC(i,20:L)));
    v12sVA(i,:) = w.*(v12sVA(i,:)-mean(v12sVA(i,20:L)));
    v12sVB(i,:) = w.*(v12sVB(i,:)-mean(v12sVB(i,20:L)));
    v12sVC(i,:) = w.*(v12sVC(i,:)-mean(v12sVC(i,20:L)));

    v8sFA(i,:) = w.*(v8sFA(i,:)-mean(v8sFA(i,:)));
    v8sFB(i,:) = w.*(v8sFB(i,:)-mean(v8sFB(i,20:L)));
```

```
v8sFC(i,:) = w.*(v8sFC(i,:)-mean(v8sFC(i,20:L)));
v8sVA(i,:) = w.*(v8sVA(i,:)-mean(v8sVA(i,20:L)));
v8sVB(i,:) = w.*(v8sVB(i,:)-mean(v8sVB(i,20:L)));
v8sVC(i,:) = w.*(v8sVC(i,:)-mean(v8sVC(i,20:L)));
% New springs
v16sFAO(i,:) = w.*(v16sFAO(i,:)-mean(v16sFAO(i,20:L)));
v16sFBO(i,:) = w.*(v16sFBO(i,:)-mean(v16sFBO(i,20:L)));
v16sFCO(i,:) = w.*(v16sFCO(i,:)-mean(v16sFCO(i,20:L)));
v16sVAO(i,:) = w.*(v16sVAO(i,:)-mean(v16sVAO(i,20:L)));
v16sVBO(i,:) = w.*(v16sVBO(i,:)-mean(v16sVBO(i,20:L)));
v16sVCO(i,:) = w.*(v16sVCO(i,:)-mean(v16sVCO(i,20:L)));

v12sFAO(i,:) = w.*(v12sFAO(i,:)-mean(v12sFAO(i,20:L)));
v12sFBO(i,:) = w.*(v12sFBO(i,:)-mean(v12sFBO(i,20:L)));
v12sFCO(i,:) = w.*(v12sFCO(i,:)-mean(v12sFCO(i,20:L)));
v12sVAO(i,:) = w.*(v12sVAO(i,:)-mean(v12sVAO(i,20:L)));
v12sVBO(i,:) = w.*(v12sVBO(i,:)-mean(v12sVBO(i,20:L)));
v12sVCO(i,:) = w.*(v12sVCO(i,:)-mean(v12sVCO(i,20:L)));

v8sFAO(i,:) = w.*(v8sFAO(i,:)-mean(v8sFAO(i,20:L)));
v8sFBO(i,:) = w.*(v8sFBO(i,:)-mean(v8sFBO(i,20:L)));
v8sFCO(i,:) = w.*(v8sFCO(i,:)-mean(v8sFCO(i,20:L)));
v8sVAO(i,:) = w.*(v8sVAO(i,:)-mean(v8sVAO(i,20:L)));
v8sVBO(i,:) = w.*(v8sVBO(i,:)-mean(v8sVBO(i,20:L)));
v8sVCO(i,:) = w.*(v8sVCO(i,:)-mean(v8sVCO(i,20:L)));
% Get the Fourier transform
P16sFA(i,:) = abs(fft(v16sFA(i,:)));
P16sFB(i,:) = abs(fft(v16sFB(i,:)));
P16sFC(i,:) = abs(fft(v16sFC(i,:)));
P16sVA(i,:) = abs(fft(v16sVA(i,:)));
P16sVB(i,:) = abs(fft(v16sVB(i,:)));
P16sVC(i,:) = abs(fft(v16sVC(i,:)));

P16sFAO(i,:) = abs(fft(v16sFAO(i,:)));
P16sFBO(i,:) = abs(fft(v16sFBO(i,:)));
P16sFCO(i,:) = abs(fft(v16sFCO(i,:)));
P16sVAO(i,:) = abs(fft(v16sVAO(i,:)));
P16sVBO(i,:) = abs(fft(v16sVBO(i,:)));
P16sVCO(i,:) = abs(fft(v16sVCO(i,:)));

P12sFA(i,:) = abs(fft(v12sFA(i,:)));
P12sFB(i,:) = abs(fft(v12sFB(i,:)));
P12sFC(i,:) = abs(fft(v12sFC(i,:)));
P12sVA(i,:) = abs(fft(v12sVA(i,:)));
P12sVB(i,:) = abs(fft(v12sVB(i,:)));
P12sVC(i,:) = abs(fft(v12sVC(i,:)));

P12sFAO(i,:) = abs(fft(v12sFAO(i,:)));
P12sFBO(i,:) = abs(fft(v12sFBO(i,:)));
P12sFCO(i,:) = abs(fft(v12sFCO(i,:)));
P12sVAO(i,:) = abs(fft(v12sVAO(i,:)));
P12sVBO(i,:) = abs(fft(v12sVBO(i,:)));
P12sVCO(i,:) = abs(fft(v12sVCO(i,:)));

P8sFA(i,:) = abs(fft(v8sFA(i,:)));
P8sFB(i,:) = abs(fft(v8sFB(i,:)));
```

```
P8sFC(i,:) = abs(fft(v8sFC(i,:)));
P8sVA(i,:) = abs(fft(v8sVA(i,:)));
P8sVB(i,:) = abs(fft(v8sVB(i,:)));
P8sVC(i,:) = abs(fft(v8sVC(i,:)));

P8sFAO(i,:) = abs(fft(v8sFAO(i,:)));
P8sFBO(i,:) = abs(fft(v8sFBO(i,:)));
P8sFCO(i,:) = abs(fft(v8sFCO(i,:)));
P8sVAO(i,:) = abs(fft(v8sVAO(i,:)));
P8sVBO(i,:) = abs(fft(v8sVBO(i,:)));
P8sVCO(i,:) = abs(fft(v8sVCO(i,:)));
% Check the 2 second interval
v12sVCh1(i,:) = wch.*(v12sVCh1(i,:)-mean(v12sVCh1(i,20:Lch)));
P12sVCh1(i,:) = abs(fft(v12sVCh1(i,:)));
[MP12VCh1(i),L12VCh1(i)] = max(P12sVCh1(i,1:Lch/2));
v12sVCh2(i,:) = wch.*(v12sVCh2(i,:)-mean(v12sVCh2(i,20:Lch)));
P12sVCh2(i,:) = abs(fft(v12sVCh2(i,:)));
[MP12VCh2(i),L12VCh2(i)] = max(P12sVCh2(i,1:Lch/2));
v12sVCh3(i,:) = wch.*(v12sVCh3(i,:)-mean(v12sVCh3(i,20:Lch)));
P12sVCh3(i,:) = abs(fft(v12sVCh3(i,:)));
[MP12VCh3(i),L12VCh3(i)] = max(P12sVCh3(i,1:Lch/2));
v12sVCh4(i,:) = wch.*(v12sVCh4(i,:)-mean(v12sVCh4(i,20:Lch)));
P12sVCh4(i,:) = abs(fft(v12sVCh4(i,:)));
[MP12VCh4(i),L12VCh4(i)] = max(P12sVCh4(i,1:Lch/2));
v12sVCh5(i,:) = wch.*(v12sVCh5(i,:)-mean(v12sVCh5(i,20:Lch)));
P12sVCh5(i,:) = abs(fft(v12sVCh5(i,:)));
[MP12VCh5(i),L12VCh5(i)] = max(P12sVCh5(i,1:Lch/2));
% Get 12 sided dirty FFT
 P12sVB2(i,:) = abs(fft(v12sVB2(i,:)));
% Get maximum power location and magnitude
for j = 1:3
    if j == 1
        [MP16F(i,j),L16F(i,j)] = max(P16sFA(i,1:L/2));
        [MP16V(i,j),L16V(i,j)] = max(P16sVA(i,1:L/2));
        [MP16FO(i,j),L16FO(i,j)] = max(P16sFAO(i,1:L/2));
        [MP16VO(i,j),L16VO(i,j)] = max(P16sVAO(i,1:L/2));

        [MP12F(i,j),L12F(i,j)] = max(P12sFA(i,1:L/2));
        [MP12V(i,j),L12V(i,j)] = max(P12sVA(i,1:L/2));
        [MP12FO(i,j),L12FO(i,j)] = max(P12sFAO(i,1:L/2));
        [MP12VO(i,j),L12VO(i,j)] = max(P12sVAO(i,1:L/2));

        [MP8F(i,j),L8F(i,j)] = max(P8sFA(i,1:L/2));
        [MP8V(i,j),L8V(i,j)] = max(P8sVA(i,1:L/2));
        [MP8FO(i,j),L8FO(i,j)] = max(P8sFAO(i,1:L/2));
        [MP8VO(i,j),L8VO(i,j)] = max(P8sVAO(i,1:L/2));
    elseif j == 2
        [MP16F(i,j),L16F(i,j)] = max(P16sFB(i,1:L/2));
        [MP16V(i,j),L16V(i,j)] = max(P16sVB(i,1:L/2));
        [MP16FO(i,j),L16FO(i,j)] = max(P16sFBO(i,1:L/2));
        [MP16VO(i,j),L16VO(i,j)] = max(P16sVBO(i,1:L/2));

        [MP12F(i,j),L12F(i,j)] = max(P12sFB(i,1:L/2));
        [MP12V(i,j),L12V(i,j)] = max(P12sVB(i,1:L/2));
        [MP12FO(i,j),L12FO(i,j)] = max(P12sFBO(i,1:L/2));
        [MP12VO(i,j),L12VO(i,j)] = max(P12sVBO(i,1:L/2));
```

```
            [MP8F(i,j),L8F(i,j)] = max(P8sFB(i,1:L/2));
            [MP8V(i,j),L8V(i,j)] = max(P8sVB(i,1:L/2));
            [MP8FO(i,j),L8FO(i,j)] = max(P8sFBO(i,1:L/2));
            [MP8VO(i,j),L8VO(i,j)] = max(P8sVBO(i,1:L/2));
        else
            [MP16F(i,j),L16F(i,j)] = max(P16sFC(i,1:L/2));
            [MP16V(i,j),L16V(i,j)] = max(P16sVC(i,1:L/2));
            [MP16FO(i,j),L16FO(i,j)] = max(P16sFCO(i,1:L/2));
            [MP16VO(i,j),L16VO(i,j)] = max(P16sVCO(i,1:L/2));

            [MP12F(i,j),L12F(i,j)] = max(P12sFC(i,1:L/2));
            [MP12V(i,j),L12V(i,j)] = max(P12sVC(i,1:L/2));
            [MP12FO(i,j),L12FO(i,j)] = max(P12sFCO(i,1:L/2));
            [MP12VO(i,j),L12VO(i,j)] = max(P12sVCO(i,1:L/2));

            [MP8F(i,j),L8F(i,j)] = max(P8sFC(i,1:L/2));
            [MP8V(i,j),L8V(i,j)] = max(P8sVC(i,1:L/2));
            [MP8FO(i,j),L8FO(i,j)] = max(P8sFCO(i,1:L/2));
            [MP8VO(i,j),L8VO(i,j)] = max(P8sVCO(i,1:L/2));
        end
    end
end

% Set up dimension grid
X = [-9, 0, 9];
for j = 1:5
    for k = 1:3
        Y8(j,k) = -d(j)/(2*rf8(k));
        Y12(j,k) = -d(j)/(2*rf12(k));
        Y16(j,k) = -d(j)/(2*rf16(k));
    end
end

for g = 1:5
    for h = 1:3
        if h == 2
            Z16F(g,h) = f(L16F(g,h));%*2*rf16(h)/V;
        else
            Z16F(g,h) = f(L16F(g,h))/2;%*rf16(h)/V;
        end
        Z16V(g,h) = f(L16V(g,h))/2;%*rf16(h)/V;

        Z16FO(g,h) = f(L16FO(g,h))/2;%*rf16(h)/V;
        Z16VO(g,h) = f(L16VO(g,h))/2;%*rf16(h)/V;

        Z12F(g,h) = f(L12F(g,h))/2;%*rf12(h)/V;
        Z12V(g,h) = f(L12V(g,h))/2;%*rf12(h)/V;
        Z12FO(g,h) = f(L12FO(g,h))/2;%*rf12(h)/V;
        Z12VO(g,h) = f(L12VO(g,h))/2;%*rf12(h)/V;

        Z8F(g,h) = f(L8F(g,h))/2;%*rf8(h)/V;
        Z8V(g,h) = f(L8V(g,h))/2;%*rf8(h)/V;
        Z8FO(g,h) = f(L8FO(g,h))/2;%*rf8(h)/V;
        Z8VO(g,h) = f(L8VO(g,h))/2;%*rf8(h)/V;
```

```
    end
    Z12VCh1(g) = fch(L12VCh1(g))/2;
    Z12VCh2(g) = fch(L12VCh2(g))/2;
    Z12VCh3(g) = fch(L12VCh3(g))/2;
    Z12VCh4(g) = fch(L12VCh4(g))/2;
    Z12VCh5(g) = fch(L12VCh5(g))/2;
end
Z8FO(3,1) = 0.01; % Just noise, no clear peak

figure(1)
plot(Y8(:,1),Z8F(:,1),'sqb')
hold on
plot(Y8(:,2),Z8F(:,2),'ob')
plot(Y8(:,3),Z8F(:,3),'*b')

plot(Y8(:,1),Z8FO(:,1),'sqk')
plot(Y8(:,2),Z8FO(:,2),'ok')
plot(Y8(:,3),Z8FO(:,3),'*k')

grid on
legend('-9" w/o Oscillation','0" w/o Oscillation','9" w/o Oscillation',...
    '-9" w/ Oscillation','0" w/ Oscillation','9" w/ Oscillation',3)
xlabel('Cross flow position from center of model (Model local diameters)')
ylabel('Frequency (Hz)')
title('8 sided model, Face upwind configuration')

figure(2)
plot(Y8(:,1),Z8V(:,1),'sqb')
hold on
plot(Y8(:,2),Z8V(:,2),'ob')
plot(Y8(:,3),Z8V(:,3),'*b')

plot(Y8(:,1),Z8VO(:,1),'sqk')
plot(Y8(:,2),Z8VO(:,2),'ok')
plot(Y8(:,3),Z8VO(:,3),'*k')

grid on
legend('-9" w/o Oscillation','0" w/o Oscillation','9" w/o Oscillation',...
    '-9" w/ Oscillation','0" w/ Oscillation','9" w/ Oscillation',3)
xlabel('Cross flow position from center of model (in)')
ylabel('Frequency (Hz)')
title('8 sided model, Vertex upwind configuration')

figure(3)
plot(Y12(:,1),Z12F(:,1),'sqb')
hold on
plot(Y12(:,2),Z12F(:,2),'ob')
plot(Y12(:,3),Z12F(:,3),'*b')

plot(Y12(:,1),Z12FO(:,1),'sqr')
plot(Y12(:,2),Z12FO(:,2),'or')
plot(Y12(:,3),Z12FO(:,3),'*r')

grid on
legend('-9" w/o Oscillation','0" w/o Oscillation','9" w/o Oscillation',...
```

```
    '-9" w/ Oscillation','0" w/ Oscillation','9" w/ Oscillation',3)
xlabel('Cross flow position from center of model (Model local diameters)')
ylabel('Frequency (Hz)')
title('12 sided model, Face upwind configuration')

figure(4)
plot(Y12(:,1),Z12V(:,1),'sqb')
hold on
plot(Y12(:,2),Z12V(:,2),'ob')
plot(Y12(:,3),Z12V(:,3),'*b')

plot(Y12(:,1),Z12VO(:,1),'sqr')
plot(Y12(:,2),Z12VO(:,2),'or')
plot(Y12(:,3),Z12VO(:,3),'*r')

grid on
legend('-9" w/o Oscillation','0" w/o Oscillation','9" w/o Oscillation',...
    '-9" w/ Oscillation','0" w/ Oscillation','9" w/ Oscillation',3)
xlabel('Cross flow position from center of model (Model local diameters)')
ylabel('Frequency (Hz)')
title('12 sided model, Vertex upwind configuration')

figure(5)
plot(Y16(:,1),Z16F(:,1),'sqb')
hold on
plot(Y16(:,2),Z16F(:,2),'ob')
plot(Y16(:,3),Z16F(:,3),'*b')

plot(Y16(:,1),Z16FO(:,1),'sqk')
plot(Y16(:,2),Z16FO(:,2),'ok')
plot(Y16(:,3),Z16FO(:,3),'*k')

grid on
legend('-9" w/o Oscillation','0" w/o Oscillation','9" w/o Oscillation',...
    '-9" w/ Oscillation','0" w/ Oscillation','9" w/ Oscillation',3)
xlabel('Cross flow position from center of model (Model local diameters)')
ylabel('Frequency (Hz)')
title('16 sided model, Face upwind configuration')

figure(6)
plot(Y16(:,1),Z16V(:,1),'sqb')
hold on
plot(Y16(:,2),Z16V(:,2),'ob')
plot(Y16(:,3),Z16V(:,3),'*b')

plot(Y16(:,1),Z16VO(:,1),'sqk') % 1 is fine, it was empty data
plot(Y16(:,2),Z16VO(:,2),'ok')
plot(Y16(:,3),Z16VO(:,3),'*k')

grid on
legend('-9" w/o Oscillation','0" w/o Oscillation','9" w/o Oscillation',...
    '-9" w/ Oscillation','0" w/ Oscillation','9" w/ Oscillation',3)
xlabel('Cross flow position from center of model (Model local diameters)')
ylabel('Frequency (Hz)')
title('16 sided model, Vertex upwind configuration')
```

```
figure(7)
hold on
plot(Y8(:,1),Z8V(:,1),'sqr')
plot(Y8(:,2),Z8V(:,2),'or')
plot(Y8(:,3),Z8V(:,3),'*r')

plot(Y12(:,1),Z12V(:,1),'sqg')
plot(Y12(:,2),Z12V(:,2),'og')
plot(Y12(:,3),Z12V(:,3),'*g')

plot(Y16(:,1),Z16V(:,1),'sqm')
plot(Y16(:,2),Z16V(:,2),'om')
plot(Y16(:,3),Z16V(:,3),'*m')

grid on
legend('8s, -9"','8s, 0"','8s, 9"','12s, -9"','12s, 0"','12s, 9"',...
    '16s, -9"','16s, -0"','16s, 9"',3)
xlabel('Cross flow position from center of model (Model local diameters)')
ylabel('Frequency (Hz)')
title('3 types of model, Vertex upwind, no oscillation')

figure(8)
hold on
plot(Y8(:,1),Z8VO(:,1),'sqr')
plot(Y8(:,2),Z8VO(:,2),'or')
plot(Y8(:,3),Z8VO(:,3),'*r')

plot(Y12(:,1),Z12VO(:,1),'sqg')
plot(Y12(:,2),Z12VO(:,2),'og')
plot(Y12(:,3),Z12VO(:,3),'*g')

plot(Y16(:,1),Z16VO(:,1),'sqm')
plot(Y16(:,2),Z16VO(:,2),'om')
plot(Y16(:,3),Z16VO(:,3),'*m')

grid on
legend('8s, -9"','8s, 0"','8s, 9"','12s, -9"','12s, 0"','12s, 9"',...
    '16s, -9"','16s, -0"','16s, 9"',3)
xlabel('Cross flow position from center of model (Model local diameters)')
ylabel('Frequency (Hz)')
title('3 types of model, Vertex upwind, with oscillation')

figure(9)
plot([1 2 3 4 5],Z12VCh1,'bsq')
hold on
plot([1 2 3 4 5],Z12VCh2,'om')
plot([1 2 3 4 5],Z12VCh3,'*g')
plot([1 2 3 4 5],Z12VCh4,'rsq')
plot([1 2 3 4 5],Z12VCh5,'cy^')
grid on
legend('0.59 Model diameters away','0.78 Model diameters away',...
    '0.96 Model diameters away','1.14 Model diameters away',...
    '1.32 Model diameters away')
xlabel('Sample number')
```

```
ylabel('Frequency (Hz)')
title('2 second windows of a 10 second signal')
set(gca,'XTickLabel',[1 2 3 4 5])

figure(10)
subplot(2,1,1)
plot(t12sVB(5,:),v12sVB2(5,:)-mean(v12sVB2(5,20:L)))
hold on
grid on
axis([0 10 -0.02 0.02])
title('Data before post-processing')
xlabel('Time (s)')
ylabel('Voltage (V)')
subplot(2,1,2)
plot(t12sVB(5,:),v12sVB(5,:),'r')
grid on
axis([0 10 -0.02 0.02])
title('Data after post-processing')
xlabel('Time (s)')
ylabel('Voltage (V)')

figure(11)
subplot(2,1,1)
plot(f/2,P12sVB2(2,1:L/2))
axis([0 125 0 10])
grid on
title('Data before post-processing')
xlabel('Frequency (Hz)')
ylabel('Power ()')
subplot(2,1,2)
plot(f/2,P12sVB(2,1:L/2),'r')
axis([0 125 0 10])
grid on
title('Data after post-processing')
xlabel('Frequency (Hz)')
ylabel('Power ()')

%% Pressure Processing
% Jaime Ocampo
%% Load data
clc
clear all
close all

ftest1 = load('0deg.txt');
ftest2 = load('45deg.txt');
ftest3 = load('90deg.txt');
ftest4 = load('135deg.txt');
ftest5 = load('180deg.txt');

cal1 = load('Calibration.txt');
cal2 = load('Calibration2.txt');
cal3 = load('Calibration3.txt');

%% Data Analysis
q = 0.5*1.225*9.84^2*0.000145037738;
```

```
for i = 1:12
    mcal1(i) = mean(cal1(:,i));
    mcal2(i) = mean(cal2(:,i));
    mcal3(i) = mean(cal3(:,i));

    Cptest1(i) = (mean(ftest1(:,i))-mcal3(i))/q;
    Cptest2(i) = (mean(ftest2(:,i))-mcal3(i))/q;
    Cptest3(i) = (mean(ftest3(:,i))-mcal3(i))/q;
    Cptest4(i) = (mean(ftest4(:,i))-mcal3(i))/q;
    Cptest5(i) = (mean(ftest5(:,i))-mcal3(i))/q;
end
for i = 1:12
    Cpdeg(i,:) = [Cptest1(i),Cptest2(i),Cptest3(i),Cptest4(i),Cptest5(i)];
end

%% Plotter

Tap = 1:1:12;
Deg = [0,45,90,135,180];

% Dodecagon
figure(1)
plot(Tap,Cptest1,'*b','MarkerSize',12)
hold on
plot(Tap,Cptest2,'sqr','MarkerSize',12)
plot(Tap,Cptest3,'xcy','MarkerSize',12)
plot(Tap,Cptest4,'og','MarkerSize',12)
plot(Tap,Cptest5,'om','MarkerSize',12)

legend('O deg','45 deg','90 deg','135 deg','180 deg');

grid on
xlabel('Tap #')
ylabel('Cp')
title('8 sided model, Cp reading at different tap locations')

figure(2)
plot(Deg,Cpdeg(1,:),'*b','MarkerSize',12)
hold on
plot(Deg,Cpdeg(2,:),'sqr','MarkerSize',12)
plot(Deg,Cpdeg(3,:),'xcy','MarkerSize',12)
plot(Deg,Cpdeg(4,:),'og','MarkerSize',12)
plot(Deg,Cpdeg(5,:),'om','MarkerSize',12)
plot(Deg,Cpdeg(6,:),'sqk','MarkerSize',12)
plot(Deg,Cpdeg(7,:),'xb','MarkerSize',12)
Expt = [0 30 60 90 120 150 180];
Data = [1 0.2 -0.9 -1.01 -1.05 -1.07 -1.02];
plot(Expt,Data,'xk','MarkerSize',12)
% theta = 1:180;
% circ = 1-4*sin(theta*pi/180).^2;
% plot(theta,circ,'--k')
% plot(Deg,Cpdeg(8,:),'sqr','MarkerSize',10)
% plot(Deg,Cpdeg(9,:),'*cy','MarkerSize',10)
% plot(Deg,Cpdeg(10,:),'*k','MarkerSize',10)
% plot(Deg,Cpdeg(11,:),'*b','MarkerSize',10)
```

```
% plot(Deg,Cpdeg(12,:),'*g','MarkerSize',10)

grid on
xlabel('Angle (deg)')
ylabel('Cp')
title('8 sided model, Angle of model face with respect to wind, 0 is
stagnation')
legend('Tap 1','Tap 2','Tap 3','Tap 4','Tap 5','Tap 6','Tap 7','Experimental
Cylinder',...
    'Tap 9','Tap 10','Tap 11','Tap 12');

%% Pressure Processing
% Jaime Ocampo
%% Load data
clc
clear all
close all

ftest1 = load('12s0deg.txt');
ftest2 = load('12s30deg.txt');
ftest3 = load('12s60deg.txt');
ftest4 = load('12s90deg.txt');
ftest5 = load('12s120deg.txt');
ftest6 = load('12s150deg.txt');
ftest7 = load('12s180deg.txt');

cal1 = load('12sCalibration1.txt');
cal2 = load('12sCalibration2.txt');
cal3 = load('Calibration12s.txt');

%% Data Analysis
q = 0.5*1.225*9.84^2*0.000145037738;
for i = 1:12
    mcal1(i) = mean(cal1(:,i));
    mcal2(i) = mean(cal2(:,i));
    mcaltot(i) = (mcal1(i)+mcal2(i))/2;
    mcal3(i) = mean(cal3(:,i));

    Cptest1(i)                        =                    (mean(ftest1(:,i))-
mcal3(i))/(0.5*1.225*9.89^2*0.000145037738);
    Cptest2(i) = (mean(ftest2(:,i))-mcaltot(i))/q;
    Cptest3(i) = (mean(ftest3(:,i))-mcaltot(i))/q;
    Cptest4(i) = (mean(ftest4(:,i))-mcaltot(i))/q;
    Cptest5(i) = (mean(ftest5(:,i))-mcaltot(i))/q;
    Cptest6(i) = (mean(ftest6(:,i))-mcaltot(i))/q;
    Cptest7(i) = (mean(ftest7(:,i))-mcaltot(i))/q;
end
for i = 1:12
    Cpdeg(i,:) = [Cptest1(i),Cptest2(i),Cptest3(i),Cptest4(i),Cptest5(i),...
        Cptest6(i),Cptest7(i)];
end

%% Plotter

Tap = 1:1:12;
```

```
Deg = [0,30,60,90,120,150,180];

% Dodecagon
figure(1)
plot(Tap,Cptest1,'*b','MarkerSize',12)
hold on
plot(Tap,Cptest2,'sqr','MarkerSize',12)
plot(Tap,Cptest3,'xcy','MarkerSize',12)
plot(Tap,Cptest4,'og','MarkerSize',12)
plot(Tap,Cptest5,'om','MarkerSize',12)
plot(Tap,Cptest6,'sqk','MarkerSize',12)
plot(Tap,Cptest7,'xb','MarkerSize',12)

legend('O deg','30 deg','60 deg','90 deg','120 deg','150 deg','180 deg');

grid on
xlabel('Tap #')
ylabel('Cp')
title('12 sided model, Cp reading at different tap locations')

figure(2)
plot(Deg,Cpdeg(1,:),'*b','MarkerSize',12)
hold on
plot(Deg,Cpdeg(2,:),'sqr','MarkerSize',12)
plot(Deg,Cpdeg(3,:),'xcy','MarkerSize',12)
plot(Deg,Cpdeg(4,:),'og','MarkerSize',12)
plot(Deg,Cpdeg(5,:),'om','MarkerSize',12)
plot(Deg,Cpdeg(6,:),'sqk','MarkerSize',12)
plot(Deg,Cpdeg(7,:),'xb','MarkerSize',12)
plot(Deg,Cpdeg(8,:),'sqr','MarkerSize',10)
plot(Deg,Cpdeg(9,:),'*cy','MarkerSize',10)
plot(Deg,Cpdeg(10,:),'*k','MarkerSize',10)
plot(Deg,Cpdeg(11,:),'*b','MarkerSize',10)
plot(Deg,Cpdeg(12,:),'*g','MarkerSize',10)
Expt = [0 30 60 90 120 150 180];
Data = [1 0.2 -0.9 -1.01 -1.05 -1.07 -1.02];
plot(Expt,Data,'xk','MarkerSize',12)
% theta = 1:180;
% circ = 1-4*sin(theta*pi/180).^2;
% plot(theta,circ,'--k')

grid on
xlabel('Angle (deg)')
ylabel('Cp')
title('12 sided model, Angle of model face with respect to wind, 0 is
stagnation')
legend('Tap 1','Tap 2','Tap 3','Tap 4','Tap 5','Tap 6','Tap 7','Tap 8',...
    'Tap 9','Tap 10','Tap 11','Tap 12','Theoretical Cylinder');

%% Pressure Processing
% Jaime Ocampo
%% Load data
clc
clear all
close all
```

```
ftest1 = load('16s0deg.txt');
ftest2 = load('16s22_5deg.txt');
ftest3 = load('16s45deg.txt');
ftest4 = load('16s67_5deg.txt');
ftest5 = load('16s90deg.txt');
ftest6 = load('16s112_5deg.txt');
ftest7 = load('16s135deg.txt');
ftest8 = load('16s157_5deg.txt');
ftest9 = load('16s180deg.txt');

cal1 = load('Calibration1.txt');
cal2 = load('Calibration2.txt');


%% Data Analysis
q = 0.5*1.225*9.89^2*0.000145037738;
for i = 1:12
    mcal1(i) = mean(cal1(:,i));
    mcal2(i) = mean(cal2(:,i));
    mcaltot(i) = (mcal1(i)+mcal2(i))/2;

    Cptest1(i) = (mean(ftest1(:,i))-mcaltot(i))/q;
    Cptest2(i) = (mean(ftest2(:,i))-mcaltot(i))/q;
    Cptest3(i) = (mean(ftest3(:,i))-mcaltot(i))/q;
    Cptest4(i) = (mean(ftest4(:,i))-mcaltot(i))/q;
    Cptest5(i) = (mean(ftest5(:,i))-mcaltot(i))/q;
    Cptest6(i) = (mean(ftest6(:,i))-mcaltot(i))/q;
    Cptest7(i) = (mean(ftest7(:,i))-mcaltot(i))/q;
    Cptest8(i) = (mean(ftest8(:,i))-mcaltot(i))/q;
    Cptest9(i) = (mean(ftest9(:,i))-mcaltot(i))/q;
end
for i = 1:12
    Cpdeg(i,:) = [Cptest1(i),Cptest2(i),Cptest3(i),Cptest4(i),Cptest5(i),...
        Cptest6(i),Cptest7(i),Cptest8(i),Cptest9(i)];
end

%% Plotter

Tap = 1:1:12;
Deg = [0,22.5,45,67.5,90,112.5,135,157.5,180];

% Dodecagon
figure(1)
plot(Tap,Cptest1,'*b','MarkerSize',12)
hold on
plot(Tap,Cptest2,'sqr','MarkerSize',12)
plot(Tap,Cptest3,'xcy','MarkerSize',12)
plot(Tap,Cptest4,'og','MarkerSize',12)
plot(Tap,Cptest5,'om','MarkerSize',12)
plot(Tap,Cptest6,'sqk','MarkerSize',12)
plot(Tap,Cptest7,'xb','MarkerSize',12)
plot(Tap,Cptest8,'*r','MarkerSize',12)
plot(Tap,Cptest9,'*b','MarkerSize',12)
```

```
legend('O deg','22.5 deg','45 deg','67.5 deg','90 deg','112.5 deg',...
    '135 deg','157.5 deg','180 deg');

grid on
xlabel('Tap #')
ylabel('Cp')
title('16 sided model, Cp reading at different tap locations')

figure(2)
plot(Deg,Cpdeg(1,:),'*b','MarkerSize',12)
hold on
plot(Deg,Cpdeg(2,:),'sqr','MarkerSize',12)
plot(Deg,Cpdeg(3,:),'xcy','MarkerSize',12)
plot(Deg,Cpdeg(4,:),'og','MarkerSize',12)
plot(Deg,Cpdeg(5,:),'om','MarkerSize',12)
plot(Deg,Cpdeg(6,:),'sqk','MarkerSize',12)
plot(Deg,Cpdeg(7,:),'xb','MarkerSize',12)
plot(Deg,Cpdeg(8,:),'sqr','MarkerSize',10)
plot(Deg,Cpdeg(9,:),'*cy','MarkerSize',10)
plot(Deg,Cpdeg(10,:),'*k','MarkerSize',10)
plot(Deg,Cpdeg(11,:),'*b','MarkerSize',10)
plot(Deg,Cpdeg(12,:),'*g','MarkerSize',10)
Expt = [0 30 60 90 120 150 180];
Data = [1 0.2 -0.9 -1.01 -1.05 -1.07 -1.02];
plot(Expt,Data,'xk','MarkerSize',12)
% theta = 1:180;
% circ = 1-4*sin(theta*pi/180).^2;
% plot(theta,circ,'--k')

grid on
xlabel('Angle (deg)')
ylabel('Cp')
title('16 sided model, Angle of model face with respect to wind, 0 is
stagnation')
legend('Tap 1','Tap 2','Tap 3','Tap 4','Tap 5','Tap 6','Tap 7','Tap 8',...
    'Tap 9','Tap 10','Tap 11','Tap 12','Theoretical Cylinder');
```