Group D - Affifi Hussen, Garrett Marlowe, Terry Newsome, Pablo Torrico
Progress Report Week 5
Instructor – Dr. Kecskemety, GTA – Rahel Beyene 2/15/2016

Week 4

Backwards looking summary

Situation

The main objective of the previous lab was to determine the most effective AEV concept design. Creativity and concept designs are essential to the development of the AEV. Key objectives for the AEV design were to minimize energy usage and torque as well as the handle variance on the track. The main objectives of the AEV design would help determine the final AEV design as each person in the group would tailor their design to fulfil the objectives. At the beginning of the lab the team would have 15 minutes to individually create a design. Then after the initial 15 minutes the group would come together to brainstorm a final AEV design. After the brainstorming session, the group had a final design that maximized the possible objectives to complete the final concept.

Results & Analysis

The lab group utilized system analysis to quantify the energy costs of running the AEV. System analysis will allow the group to understand which portions of the track the AEV uses the most energy. This is due to the layout of the track as well as operational constraints that are placed on the overall goal. In addition to the overall mission concept there are a set of operational constraints that must be completed for the AEV maximize its potential. The operational constraints include certain areas on the track where the AEV must be stationary for the gate to open. If the AEV does not remain stationary and passes the gate it will end the run. The group designed a code to make the AEV run to the beginning of the first gate. Figure 1 shows the Power on the y axis versus time on the x axis. This figure is indicative of the energy that will be used for the run. This is be extremely useful for the development of the final code as the group will be able to distinguish areas in which the AEV is using the most energy. Analysis of the graph can be further broken down into an energy analysis. Figure 2 is helpful to the group because it shows areas where the most energy is being expended. Figure 2 is broken down into four distinct phases and they correspond to the program of the AEV and its energy expenditure. The figure shows that throughout the first phase there is a constant increase in power without much deviation. This corresponds to the first straight portion of the track. Consequently, the second phase of the energy diagram corresponds to the first turn that the AEV encounters. As such there are several constant spikes in energy as the AEV must compensate in the turn to keep the power of the AEV constant. The third phase shows the portion where the AEV began to coast as it began to approach the first gate. This means that there would be a very small energy expenditure in this phase as it was not using the battery to power the AEV. Finally, the fourth phase in the energy diagram shows the part where the AEV was not moving and thus not using any power. Overall, the total amount of energy used to reach the first gate was 3362.50 joules. When all of the phases are viewed as one it shows that the code must be programmed to maximize the coasting feature of the AEV so as to not spend any energy. On the one hand it will reduce the energy spent, and coupled with the lightweight design of the AEV it will maximize the energy/mass ratio. This will fulfill the mission concept as well as create awareness for the operational constraints that must be noted in the final program.

Further analysis is required as the final AEV design was not utilized for the performance. Looking to the future there are concrete goals that the group can create with a formulated timeline to stay on track for the final project. The final AEV design is available to the group after the completion of the progress report. Thus one of the major goals of the future will be to run repeated system analysis after runs. The team must also create a functional final program for the AEV. Operational constraints, consistency ,and energy management are the factors that will be kept in mind for the final program. Thus in the next few weeks the group must code the final program. Each teammate will share the overall work.

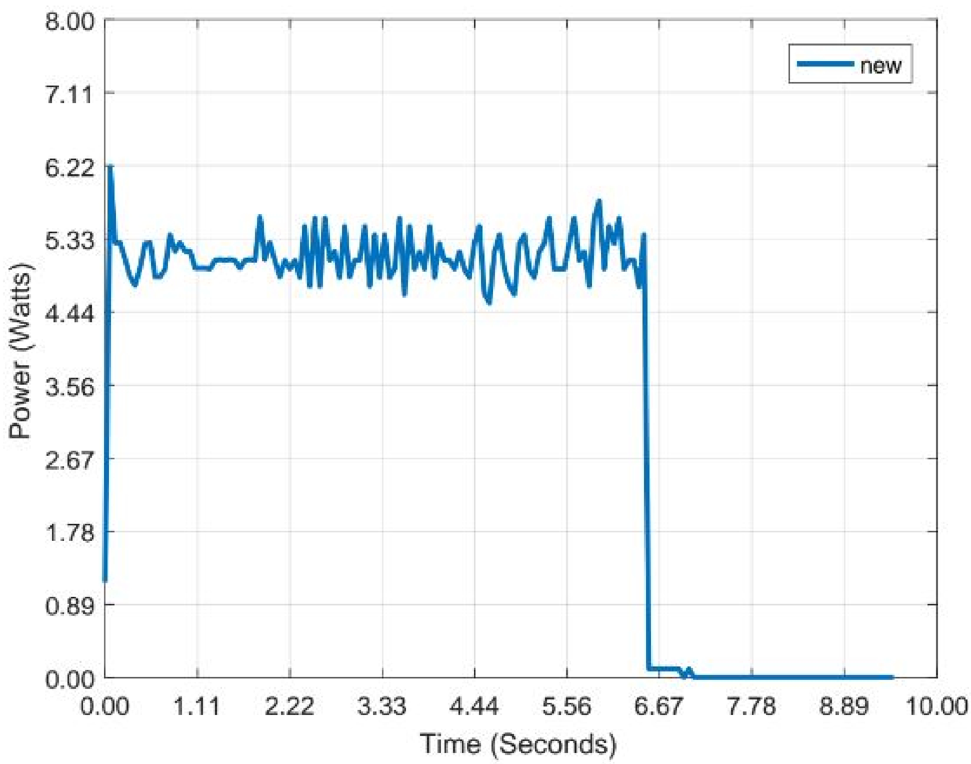

Figure 1: Power vs. Time

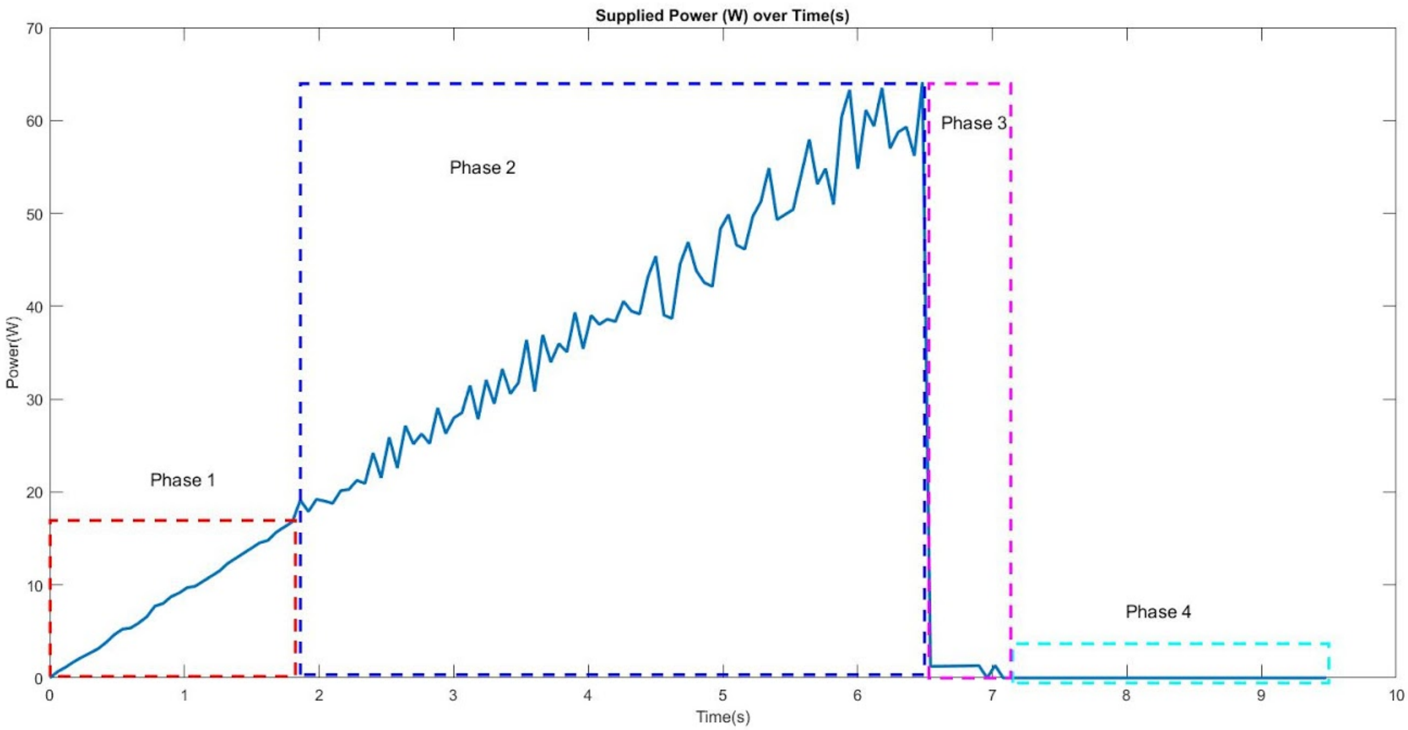

Figure 2: Energy Analysis vs. Time

Table 1: Phase Breakdown

Phase	Arduino Code	Time(seconds)	Total Energy(J)
1	P1 = Power(1:39);	0:1.8	415.729
2	P2 = Power(39:117);	1.8:6.5	2945.495
3	P3 = Power(117:128);	6.5:7.1	1.336
4	P4 = Power(128:140);	7.1:9.48	0.000
		Total Energy	3362.560

Group D - Affifi Hussen, Garrett Marlowe, Terry Newsome, Pablo Torrico
Progress Report Week 5
Instructor – Dr. Kecskemety, GTA – Rahel Beyene 2/15/2016

Takeaways

1) The team was able to download data from the automatic control system.
2) The team was able to convert EEPROM Arduino data to physical parameters.
3) The team was able to analyze performance characteristics using physical parameters.

Week 5

Situation

In observation of the next lab, the team will need to be familiar with the techniques for decision making and the structured method to screen and score design concepts. The team will achieve these objectives by programming the sample AEV for a specific operation and will be tested on the track. Using the sample AEV as a baseline reference, the team will perform concept screening and scoring methods with the AEV design concept. This lab is imperative to the team's concept design in two ways: 1) it will determine the success criteria for the AEV design, and 2) the team will know what aspects need to be improved so as to be better than the sample AEV. The sample AEV has a stable system, but it is far from being the most efficient - where the team is entitled to carry out the task to create an efficient AEV.

Weekly goals

1) Assemble the projected design for the AEV using the blueprints made so far.
2) Test and record data based on the AEV design created, and begin editing the design for possible improvements.
3) Optimize the time allocated to meet the criterias in the Mission Concept Review.

Weekly schedule

Table 1: Week 5 Schedule

Task	**Teammate(s)**	**Start Time**	**Due Date**	**Time Needed**
AEV Construction	Garrett, Drew	2-10	2-10	Complete
AEV Testing	All	-	-	-
AEV Video	Affifi , Pablo	2-10	Lab 7	Lab 7
Week 5 Progress Report	All			

Appendix

Team Meeting Notes
02-11-2017
3:00pm-4:30pm
The main objective of this meeting was to begin work on the progress report, as well as outline tasks for individual members, and construct the AEV Design.

Completed Objectives

- Tasks were assigned to individual members
- Scheduled a future meeting for the team.
- AEV Construction

Objectives to be completed

- Progress Report

Sample Calculations- Pablo Torrico Alvarez

$Time$ $t = te/1000$

$t = 120/1000$

$t = .12\ s$

where :

t= time(seconds)

te=EEPROM Time(ms)

$Current$ $I = (Ie/1024) * Vr * (1\ amp/\ 0.185\ volts)$

$I = (51/1024) * 2.46\ volts * (1\ amp/0.185\ volts)$

$I = 0.6623\ amps$

where:

I = current(amps)

Ie= EEPROM current (ADC counts)

Vr=2.46 volts

$Voltage$ $V = (15 * Ve)/1024$

$V = (15 * 545)/\ 1024$

$V = 7.983\ volts$

Where:

V=voltage(volts)

Ve= EEPROM voltage (ADC counts)

$Distance$ $d = 0.0124 * marks$

$d = 0.0124 * 0$

$d = 0\ m$

Where:

d= distance (meters)

marks= wheel counts accumulated by reflectance sensors

$Position$ $s = 0.0124 * pos$

$s = 0.0124 * 0$

$s = 0\ m$

Where:

s= AEV postion (meters from starting point)

pos= wheel counts recorded by reflectance sensors

Affifi Hussen

V_R = 2.46 V

t_E= 60 ms

Current = 60 counts

Voltage = 544 counts

Time:

$$t = \frac{t_E}{1000}$$
$$= \frac{60\ ms}{1000}$$
$$= 0.06\ s$$

Current:

$$I = \frac{I_E}{1024} \times V_R \times \frac{1\ Amp}{0.185\ volts}$$
$$= \frac{60}{1024} 2.46 \times \frac{1\ Amp}{0.185\ volts}$$
$$= 0.7791\ Amp$$

Voltage:

$$V = \frac{15V_E}{1024}$$
$$= 15 \times \frac{544}{1024}$$
$$= 7.969\ volts$$

Power:

$$P = VI$$
$$= 7.969\ volts \times 0.7791\ Amp$$
$$= 6.209\ W$$

Incremental Energy:

$$E_j = \frac{P_j + P_{j+1}}{2} \times (t_{j+1} - t_j)$$
$$= \frac{6.209 + 5.287}{2} \times (0.12 - 0.06)$$
$$= 0.3449\ J$$

Sample Calculations- Garrett Marlowe

Time

$$t = te/1000$$
$$t = 901/1000$$
$$t = .901\ s$$

where :

t= time(seconds)
te=EEPROM Time(ms)

Current

$$I = (Ie/1024) * Vr * (1\ amp/\ 0.185\ volts)$$
$$I = (51/1024) * 2.46\ volts * (1\ amp/0.185\ volts)$$
$$I = 0.6623\ amps$$

where:

I = current(amps)
Ie= EEPROM current (ADC counts)
Vr=2.46 volts

Voltage

$$V = (15 * Ve)/1024$$
$$V = (15 * 545)/\ 1024$$
$$V = 7.983\ volts$$

Where:

V=voltage(volts)
Ve= EEPROM voltage (ADC counts)

Distance

$$d = 0.0124 * marks$$
$$d = 0.0124 * 4$$
$$d = 0.0495\ m$$

Where:

d= distance (meters)
marks= wheel counts accumulated by reflectance sensors

Position

$$s = 0.0124 * pos$$
$$s = 0.0124 * 4$$
$$s = 0.0495\ m$$

Where:

s= AEV postion (meters from starting point)

Sample Calculations: Drew Newsome

Time

t= te/1000

t= 140/1000

t = .14 s

where:

t= time(seconds)

te=EEPROM Time(ms)

Current

I= (Ie/1024) * Vr*(1 amp/ 0.185 volts)

I = (55/1024) *2.46 volts * (1 amp/0.185 volts)

I= 0.5405 amps

where:

I = current(amps)

Ie= EEPROM current (ADC counts)

Vr=2.46 volts

Voltage

V= (15*Ve)/1024

V= (15*550)/ 1024

V= 8.05664

Where:

V=voltage (volts)

Ve=EEPROM voltage (ADC counts)

Distance d= 0.0124* marks

d= 0.0124* 2

d= 0.0248 m

Where:

d=distance (meters)

marks=wheel counts accumulated by reflectance sensors

Position

s= 0.0124*p

s= 0.0124* 2

s= 0.0248

Where:

s=AEV position (from start)
p=counts obtained via reflectance sensor

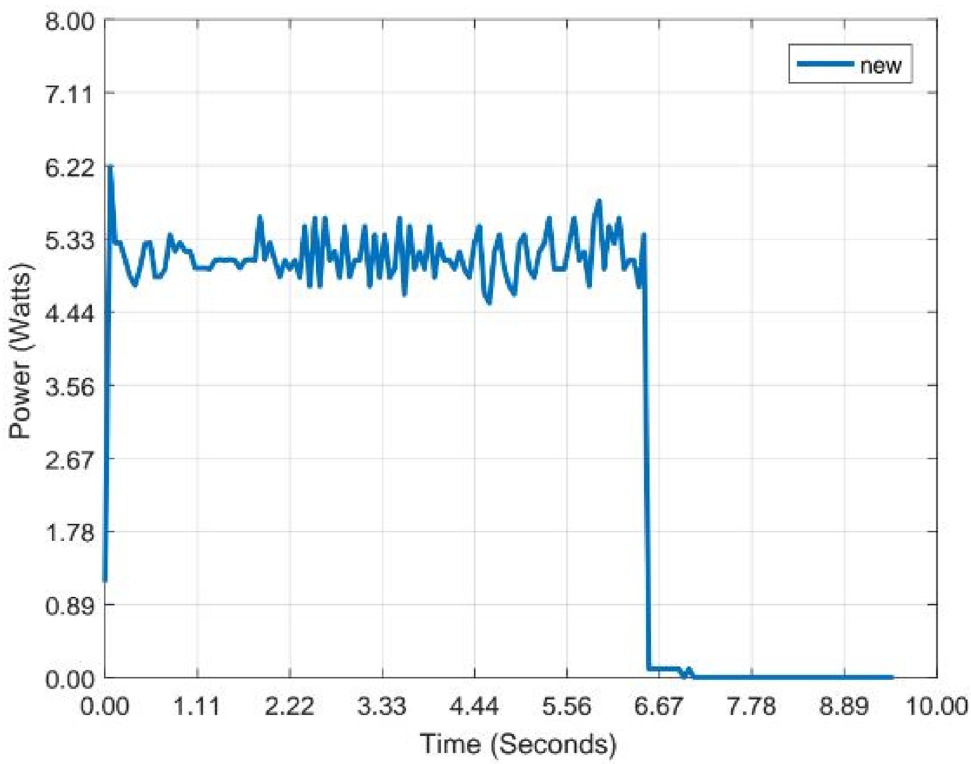

Figure 1: Power vs Time

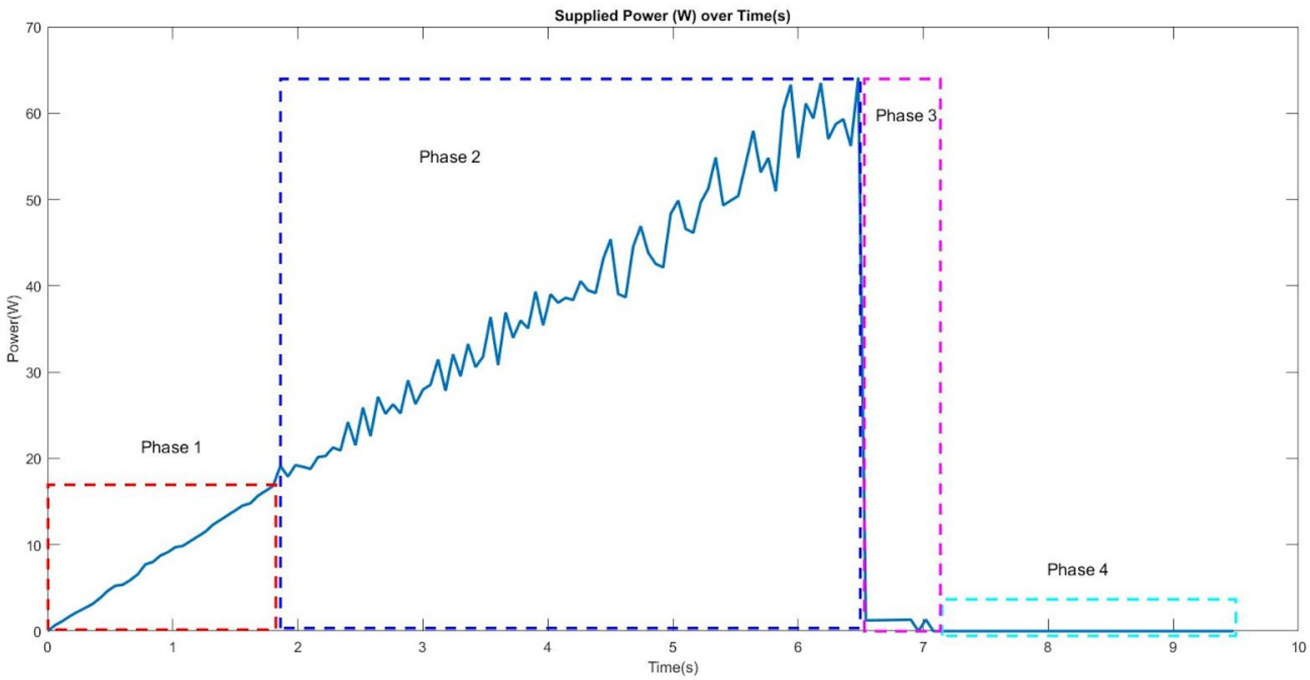

Figure 2: Energy Analysis vs Time

Table 1: Phase Breakdown

Phase	Arduino Code	Time(seconds)	Total Energy(J)
1	P1 = Power(1:39);	0:1.8	415.729
2	P2 = Power(39:117);	1.8:6.5	2945.495
3	P3 = Power(117:128);	6.5:7.1	1.336
4	P4 = Power(128:140);	7.1:9.48	0.000
		Total Energy	3362.560

%Garrett Marlowe
%2/15/17

```
%1882 Dr. K 10:20
clc
clear
%Performance analysis

%loads excel
data= xlsread('lab04A');
%prep for martix
L= length(data)-6;
convertion=0;
%uses for loop to load necesary data into matrix
for i= 1:L
        j=i+6;
  convertion(i,1)=(data(j,1));
  convertion(i,2)=(data(j,2));
  convertion(i,3)=(data(j,3)); %#ok<*SAGROW> %<- suppresses errors
end
%uses for loop to convert data into usable parameters
for k= 1: L
    Voltage(k)=((convertion(k,1)*15)/1024);
        Current(k)=((convertion(k,2))/1024)*2.46*(1/0.185);
        Power(k)= Voltage(k)*Current(k);
        Time(k)=convertion(k,1)/1000;
end

%Create figure
figure1 = figure;

% Create axes
axes1 = axes('Parent',figure1);
hold(axes1,'on');

% Create plot
plot(Time,Power,'LineWidth',2);

% Create xlabel
xlabel({'Time(s)'});

% Create title
title({'Supplied Power (W) over Time(s)'});

% Create ylabel
```

```
ylabel({'Power(W)'});

box(axes1,'on');
% Create phase 1
annotation(figure1,'rectangle',...
    [0.130441624365482 0.112185686653772 0.141131979695432 0.195357833655706],...
    'Color',[1 0 0],...
    'LineWidth',3,...
    'LineStyle','--');

% Create phase 2
annotation(figure1,'rectangle',...
    [0.684433164128596 0.104081632653061 0.182049069373943
0.0489795918367347],...
    'Color',[0 1 1],...
    'LineWidth',3,...
    'LineStyle','--');

% Create phase 3
annotation(figure1,'rectangle',...
    [0.636209813874788 0.110204081632653 0.0471235194585448 0.74425601226406],...
    'Color',[1 0 1],...
    'LineWidth',3,...
    'LineStyle','--');

% Create phase 4
annotation(figure1,'rectangle',...
    [0.274479166666667 0.114285714285714 0.359192576142132 0.740174379610999],...
    'Color',[0 0 1],...
    'LineWidth',3,...
    'LineStyle','--');

% Create textbox 1
annotation(figure1,'textbox',...
    [0.183682952622673 0.322769953051643 0.0475670473773264
0.0577321069272776],...
    'String',{'Phase 1'},...
    'LineStyle','none',...
    'FontSize',12,...
    'FitBoxToText','off');

% Create textbox 2
```

```
annotation(figure1,'textbox',...
	[0.356144881556684 0.714788732394366 0.0469801184433165 
0.0568506275749735],...
	'String',{'Phase 2'},...
	'LineStyle','none',...
	'FontSize',12,...
	'FitBoxToText','off');

% Create textbox 3
annotation(figure1,'textbox',...
	[0.638946806260574 0.772300469483568 0.0527198604060929 
0.0551164127622863],...
	'String',{'Phase 3'},...
	'LineStyle','none',...
	'FontSize',12,...
	'FitBoxToText','off');

% Create textbox 4
annotation(figure1,'textbox',...
	[0.744985300338409 0.169014084507042 0.052931366328257 
0.0455159528600171],...
	'String','Phase 4',...
	'LineStyle','none',...
	'FontSize',12,...
	'FitBoxToText','off');
%preps sum variables
sum1=0;
sum2=0;
sum3=0;
sum4=0;
%Total energy by increment
P1 = Power(1:39);
P2 = Power(39:117);
P3 = Power(117:128);
P4 = Power(128:140);
%Uses for loops to find the area of under the curve from the beginning of
%the phase until the end of the phase
for a = 1:length(P1)
	sum1 = sum1+P1(a);
end
for a = 1:length(P2)
	sum2= sum2+P2(a);
```

```
end
for a = 1:length(P3)
        sum3= sum3+P3(a);
end
for a = 1:length(P4)
        sum4= sum4+P4(a);
end
%Displays data with context
fprintf('The total energy in phase 1 was %.3f J\n',sum1)
fprintf('The total energy in phase 2 was %.3f J\n',sum2)
fprintf('The total energy in phase 3 was %.3f J\n',sum3)
fprintf('The total energy in phase 4 was %.3f J',sum4)
```

Published with MATLAB® R2016b