**Digital Signal Processing**

**Project 1**

**March 13th, 2019**

**By: Antoine Gannat, Aishwarya Ravishankar & Alina Suon**

*CALIFORNIA STATE UNIVERSITY LONG BEACH COLLEGE OF ENGINEERING COMPUTER ENGINEERING*

**Table of Contents**

**Part 1:**

**Problem Description.................................................................................................3**

**Solution……………………………………………………………………………..3**

**Results.......................................................................................................................3**

**Encountered Problems............................................................................................8**

**Part 2:**

**Problem Description................................................................................................9**

**Solution……………………………………………………………………………..9**

**Results........................................................................................................................9**

**Encountered Problems..............................................................................................15**

**Part 3:**

**Problem Description................................................................................................16**

**Solution……………………………………………………………………………..16**

**Results........................................................................................................................16**

**Encountered Problems..............................................................................................18**

**Part 4:**

**Problem Description................................................................................................19**

**Solution……………………………………………………………………………..19**

**Results........................................................................................................................19**

**Encountered Problems..............................................................................................25**

**Part 1: Complex Numbers**

***Problem Description: By using the equations, we used matlab as a calculator to perform operations.***

**1a.**

**By hand:**

% using the equations that were given for these complex equations we substituted and performed by hand.

x= 1-2j, y = -1+1j

a= (x+y)= 1-2j-1+1j= -1j

b= (x-y)= 1-2j+1-1j = 2-3j

c= (x\*y)= (1-2j)(-1+1j)= -1+2j+1j-2J^2= -1+3j+2= 1+3j

d= (x/y)- ((1-2j)/(-1+1j))\* ((-1-1j)/(-1-1j)) = ((-1+2j-1j+2j^2)/(1-1j+1j-1j^2))= ((-3+j)/2)= -1j+0.5j

p= x^y = (1-2j)^-1+1j= -0.4526+1.2752i

**By Matlab:**

***Solution: Results:***

|  |  |
| --- | --- |
| %We write the equations into matlab so matlab can be calculator to calculate these values.  x = 1 - 2j;  y = -1 + j;  a=(x+y);  b = (x-y);  c= (x\*y);  d= (x/y);  p= x.^y;  %these makes it so the that the output is displayed.  disp(a)  disp(b)  disp(c)  disp(d)  disp(p) | **Output:**  0.0000 - 1.0000i  2.0000 - 3.0000i  1.0000 + 3.0000i  -1.5000 + 0.5000i  -0.4526 + 1.2752i |

**1b.**

***Problem Description: Shows the collection of individual complex points on the complex plane.***

**Solution Results:**

|  |  |
| --- | --- |
| %initializes all the equations  x=1-2j;  y=-1+1j;  a = (x+y);  b = (x-y);  c = (x\*y);  d = (x/y);  p = x.^y;  hold on  %plots the a using the plot function  set(plot(real(a), imag(a)),'Marker','square')  text(real(a), imag(a), ' a');  %plots the b using the plot function  set(plot(real(b), imag(b)),'Marker','square')  text(real(b), imag(b), ' b');  %plots the c using the plot function  set(plot(real(c), imag(c)),'Marker','square')  text(real(c), imag(c), ' c');  %plots the d using the plot function  set(plot(real(d),  imag(d)),'Marker','square')  text(real(d), imag(d), ' d');  %this is the function where the plot function is stated  set(plot(real(p), imag(p)),'Marker','square')  text(real(p), imag(p), ' p');  hold off |  |

**1c.**

***Problem Description: By interpreting the representing vectors find the magnitude of each and the angle in degrees each vector lies in.***

***Solution: Results:***

|  |  |
| --- | --- |
| %this defines the equation of x and y  x=1-2j;  y=-1+1j;  %defines the equation of a,b,c,d,and p  a = (x+y);  b = (x-y);  c = (x\*y);  d = (x/y);  p = x.^y;  hold on  %using the plot this plots the values  plot([0 real(x)], [0 imag(x)]);  plot([0 real(y)], [0 imag(y)]);  plot([0 real(a)], [0 imag(a)]);  plot([0 real(b)], [0 imag(b)]);  plot([0 real(c)], [0 imag(c)]);  plot([0 real(d)], [0 imag(d)]);  plot([0 real(p)], [0 imag(p)]);  % uses the complex and imaginary  % Display the name of the points  text(real(x), imag(x), ' x');  text(real(y), imag(y), ' y');  text(real(a), imag(a), ' a');  text(real(b), imag(b), ' b');  text(real(c), imag(c), ' c');  text(real(d), imag(d), ' d');  text(real(p), imag(p), ' p');  hold off |  |

**1d.**

***Problem Description: Plot as vectors four non zero complex points using only random integer values.***

***Solution: Results:***

|  |  |
| --- | --- |
| % First quadrant  x1 = randi([1, 5]);  y1 = randi([1, 5]);  % Second quadrant  x2 = randi([1, 5]);  y2 = -randi([1, 5]);  % Third quadrant  x3 = -randi([1, 5]);  y3 = -randi([1, 5]);  % Fourth quadrant  x4 = -randi([1, 5]);  y4 = randi([1, 5]);  %using plot this plots the values  hold on;  plot([x1 x2], [y1 y2]);  plot([x2 x3], [y2 y3]);  plot([x3 x4], [y3 y4]);  plot([x4 x1], [y4 y1]);  % Get the area  area = polyarea([x1 x2 x3 x4], [y1 y2 y3 y4]);  % Display the title  title(['Area = ' num2str(area)]);  hold off; |  |

**2.**

***Problem Description: Represent the two sinusoids as phasors in the complex plane. Solution: Results:***

|  |  |
| --- | --- |
| %this is the equation for xt and yt  xt = 3\*exp(1j\*45\*pi/180);  yt = 2\*exp(1j\*(-150-90)\*pi/180);  figure(3)  %this plots the real and imaginary  plot([0,real(xt)], [0,imag(xt)], 'b-\*', 'linewidth', 2.0)  hold on  plot([0,real(yt)], [0,imag(yt)], 'r-\*', 'linewidth', 2.0);  hold off; |  |

**3.**

***Problem Description: Perform the operation on the phasors of and plot the resulting phasors.***

***Solution: Results:***

|  |  |
| --- | --- |
| %the equation for xy  xy = xt + yt;  hold on;  %plots the equation  plot([0,real(xy)], [0,imag(xy)], 'g-\*', 'linewidth', 2.0)  %finds the magintude  mag\_xt = sqrt(real(xt)^2 + imag(xt)^2);  ang\_xt = atan(imag(xt) / real(xt)) \* 180 / pi;  mag\_yt = sqrt(real(yt)^2 + imag(yt)^2);  ang\_yt = atan(imag(yt) / real(yt)) \* 180 / pi;  mag\_xy = sqrt(real(xy)^2 + imag(xy)^2);  ang\_xy = atan(imag(xy) / real(xy)) \* 180 / pi;  %plots the magnitude using polar coordinates  polar\_xt = sprintf('(%.2f, %.2f',mag\_xt,ang\_xt);  polar\_yt = sprintf('(%.2f, %.2f',mag\_yt,ang\_yt);  polar\_xy = sprintf('(%.2f, %.2f',mag\_xy,ang\_xy);  degree = [char(176) ')'];  str\_xt = strcat(polar\_xt, degree);  str\_yt = strcat(polar\_yt, degree);  str\_xy = strcat(polar\_xy, degree);  legend(str\_xt, str\_yt, str\_xy);  hold off; |  |

**4.**

***Problem Description: Using the plot determin the cosine DC offset.***

***Solution: Results:***

|  |  |
| --- | --- |
| %this examines the plot of a sinusodial function  amplitude = 1.6;  dcoffset= -0.9;  angFrequency = 15;  phaseAngle = 3.5;  %this is the function of the amplitude  fct = @(t) amplitude \* cos(angFrequency \* t) + dcoffset;  % Display the function  fplot(fct); |  |

***Encountered Problems***: ***Our main problem for this section was the calculations done by hand. We were having difficulties doing it by hand as we forgot how to use complex numbers but the matlab part was easy because matlab was a calculator for us and calculated everything that was supposed to be done.***

**Part 2: Simple Signals**

**1a.**

***Problem Description: Use figure k for k positive integer to number the plots***

***Solution: Results:***

|  |  |
| --- | --- |
| %% Part 1a  %states all the values  N = 12;  M = [4 5 7 10 15];  int = 0:2\*N-1;  xx = zeros(1,length(int));  %plots the arbitrary int  for index = 1:length(M);  xn = sin(2\*pi\*M(index)\*int/N);  figure(index);  plot(int,xn);  title('Part 2 #1'),xlabel('n'),ylabel('Amplitude')  end |  |

**1b.**

***Problem Description: Use the function to get integers P and Q.***

%% Part 1b

%to get M/N to lowest terms

[P Q] = rat(M/N)

**1c.**

***Problem Description: Use the function to print, a method of printing.***

%% Part 1c

%prints using this function

str = sprintf("Fundamental Period=%d",P);title(str)

**2.**

***Problem Description: Using the discrete time signal plot using subplot.***

***Solution: Results:***

|  |  |
| --- | --- |
| %% Part 2  %states all the values  k=[1 2 4 6];  %equations  omegak= 2\*pi\*k/5;  omegak\_int= 0:9;  %loops to keep going with length k of signals  for ii=1:length(k);  xk =  sin(omegak(ii)\*omegak\_int);  figure(5);  subplot(4,1,ii)  stem(omegak\_int,xk)  end |  |

**3.**

***Problem Description: Take in the three signals given and find the period for each term in the expression.***

***Solution: Results:***

|  |  |
| --- | --- |
| % Part 2 -3  clearvars;  %makes the step  step = 0.1;  %this plots the values over separate axis  figure(7)  Tx = 6; Ty = 4;  [m,k] = rat(Tx/Ty);  if(k\*Tx == m\*Ty)  Tz = k\*Tx;  end  %takes in the signals to find the period for each signal  n = 0:step:2\*Tz;  x\_1 = cos(2\*pi\*n/Tx) + 2\*cos(2\*pi\*n/Ty);  plot(n,x\_1,'linewidth', 2.0);  grid on  xlabel('n'); ylabel('x\_1[n]');  if(mod(Tz,pi) == 0)  title(strcat('Period = ', num2str(Tz/pi),'\pi'));  else  title(strcat('Period = ', num2str(Tz)));  end  figure(8)  Tx = 6\*pi; Ty = 4\*pi;  [m,k] = rat(Tx/Ty);  if(k\*Tx == m\*Ty)  Tz = k\*Tx;  end  n = 0:step:2\*Tz;  x\_2 = 2\*cos(2\*pi\*n/Tx) + cos(2\*pi\*n/Ty);  plot(n,x\_2,'linewidth', 2.0);  grid on  xlabel('n'); ylabel('x\_2[n]');  if(mod(Tz,pi) == 0)  title(strcat('Period = ', num2str(Tz/pi),'\pi'));  else  title(strcat('Period = ', num2str(Tz)));  end  figure(9)  Tx = 6; Ty = 24/5;  [m,k] = rat(Tx/Ty);  if(k\*Tx == m\*Ty)  Tz = k\*Tx;  end  n = 0:step:2\*Tz;  x\_3 = cos(2\*pi\*n/Tx) + 3\*sin(2\*pi\*n/Ty);  plot(n,x\_3,'linewidth', 2.0);  %labels all the axis  grid on  xlabel('n'); ylabel('x\_3[n]');  if(mod(Tz,pi) == 0)  title(strcat('Period = ', num2str(Tz/pi),'\pi'));  else  title(strcat('Period = ', num2str(Tz)));  end |  |

**4.**

***Problem Description: Plot the equations of the signals on such interval.***

***Solution: Results:***

|  |  |
| --- | --- |
| %this states the integer till 31  int\_n= 0:31;  %this is the equation of x1[n]  x1= sin(pi\*int\_n/4).\*cos(pi\*int\_n/4);  x1\_period=(sin(pi\*int\_n/4+pi\*int\_n/4)+sin(pi\*int\_n/4-pi\*int\_n/4))/2;  %this is the equation for x2  x2= (cos(pi\*int\_n/4)).^2;  %this is the equation for 3  x3= sin(pi\*int\_n/4).\*cos(pi\*int\_n/8);  x3\_period= (sin(pi\*int\_n/4+pi\*int\_n/8)+sin(pi\*int\_n/4-pi\*int\_n/8))/2;  %finds the period and plots it  figure(9)  plot(int\_n,x1)  figure(10)  plot(int\_n,x2)  figure(11)  plot(int\_n,x3) |  |

**5a.**

***Problem Description: Demonstrate the addition of two periodic signals is not always periodic by giving a counter example.***

%the addition of two periodic signals given by a counter example

sin(x)+ sin(pi(x))

x[n+kTx] + y[n+mTy]

Tx= 2pi, Ty= 2, x[n+2pi]

Tx/Ty= 2pi/2= pi/1= m/k so m =pi, k=1

**5b.**

***Problem Description: Demonstrate that the product of two periodic signals is not always periodic by giving a counter example***

%the product of two periodic signals is not always periodic by giving counter example

(sin((x+pi(x)) cos((x/pix/2))= sin((x+pi(x)/2) + (x-pi(x)/2) + sin((x+pi(x)/2) - (x-pi(x)/2))/2

= sin(x)+sin(pi(x)/2 = proving that sin(x) +sin(pi(x)) is not periodic.

***Encountered Problems***: ***In this section our main problem was trying to understand the wording of the problems. For example, on number one it literally said everything we had to do but we thought that it could not be that straight up and kept misunderstanding the wording of the problem. But with the help of our professor we could understand it better.***

**Part 3: Discrete Time Signals**

***Problem Description:*** Graph vector n with ranges -3 to 7 and another vector x with values of the signal x[n] at the given sample times. Plot the discrete time vector x[n].

**1.**

***Solution: Results:***

|  |  |
| --- | --- |
| % Values of x  x = [2 1 -1 3 0];  % x indexes  xindexes = [0 2 3 4];  % Values in the x axis  xaxis = [];  % Values in the y axis  yaxis = [];  % for n = -3 to n = 7  for n = (-3:7)  % Add n to x-axis array  xaxis = [xaxis, n];  % Add search y value for n  foundAt = find(xindexes == n);  % If found, set y to the value found in x  if (foundAt)  y = x(find(xindexes == n));  % If not found, set y = 0  else  y = 0;  end  % Add the y value to the y-axis array  yaxis = [yaxis, y];  end  % Draw the graph  stem(xaxis, yaxis); |  |

**2. *Problem Description:*** Define the y[n] vectors that corresponds with the ny vector. To get the signals to a shifted version of signal x[n] we modify the sample index vector n. This will reflect the shifting process. The sigshift function is utilized to shift vector x with index n by k.

***Solution: Results:***

|  |  |
| --- | --- |
| function [y,m] = sigshift(x, n, k)  % for each possibilities of k  switch k  case 1  xindex = n -2;  case 2  xindex = n + 1;  case 3  xindex = -n;  case 4  xindex = -n + 1;  end  % Check if xindex is out of range  if (xindex >= 1 & xindex < length(x))  % if not, get x[n]  y = x(xindex);  else  % if so, set y = 0  y = 0;  end  % save xindex to m  m = xindex;  end | No output. |

**3. *Problem Description:*** Graph a vector, y\_k[n]. Utilizing subplot and stem, we plot x[n] and y[n] for k=1 over the interval -10 to 10.

***Solution: Results:***

|  |  |
| --- | --- |
| % Values of x  x = [2 1 -1 3 0];  % Signals name  signals = ["y1[n] = x[n -2]" "y2[n] = x[n + 1]" "y3[n] = x[-n]" "y4[n] = x[-n + 1]"];  hold on;  % For k = 1 to k = 4  for k = (1:4)  % Create a subplot  subplot(4,1,k);  % Initialize the axis values  xaxis = [];  yaxis = [];  % for n = -10 to n = 10  for n = (-10:10)  % Add n to the x axis  xaxis = [xaxis, n];  % Call the sigshift function  [y, m] = sigshift(x,n,k);  % Add the result to the y axis  yaxis = [yaxis, y];  end  % Display the graph  stem(xaxis, yaxis);  % Set the x label  xlabel("n");  % Set the y label  ylabel("x[n]");  % Set the graph title  title(signals(k));  end  hold off; |  |

***Encountered Problems***: ***We had to determine the LSI property was violated. In determining so we had to incorportate imseg, stepseq, shiftseg, ect to help. There were some trouble confirming the results on the graphs.*Part 4: Linear Shift Invariant (LSI) Systems**

**1a.**

***Problem Description:*** The system given is not linear. The signals given are used to show that y[n] is violates linearity. We show the right hand side and the left hand side of the linearity equation over the span of 0 to 10 to show that it is not the same.

***Solution: Results:***

|  |  |
| --- | --- |
| %sets a and b  clearvars;  close all;  A = 2; B = -1;  %delta function of x\_1[n]  [x\_1,n] = impseq(0,0,10);  %delta function = x\_2[n]  x\_2 = 2\*x\_1;  lhs = sin((pi/2) \* (A\*x\_1 + B\*x\_2))  rhs = A\*sin((pi/2) \* x\_1) + B\*sin((pi/2) \* x\_2)  %plots all the different functions  figure(1)  subplot (1,2,1)  stem(n,lhs, 'filled','markersize',5);  axis([0 10 0 2])  xlabel('n'); ylabel('x[n]');  title('LHS: y[Ax+1[n] + Bx\_2[n] ]');  grid minor  %2 graphs of left and right hand side  subplot (1,2,2)  stem(n,rhs, 'filled','markersize',5);  axis([0 10 0 2])  xlabel('n'); ylabel('x[n]');  title('LHS: Ay[x\_1[n] + By[x\_2[n] ]');  grid minor |  |

**1b.**

***Problem Description:*** The system is not shift invariant. In being shift invariance the order of transformation and shifting is inevitable. In determining that it violates a given LSI property by using a shift amount k=1.

***Solution: Results:***

|  |  |
| --- | --- |
| clc;  clear all;  close all;  %defines what its shifting with  shift = -1;  z = @(u,w)w.\*u;  %states the interval  [u1, w1] = impseq(0,0,10);  z1 = z(w1, u1);  [z2, w2]= sigshift(z1,w1,shift);  %defines the shift  [u2,w3] = sigshift(u1, w1, shift);  z3 = z(u2,w3);  %this plots over the axis  subplot(2,1,1);  stem(w2, z2);  %gives the title  title('z(wu(w)-1)');  subplot(2,1,2);  stem(w3, z3);  title('z((w-1)u(w-1))'); | https://scontent-lax3-1.xx.fbcdn.net/v/t1.15752-9/52599015_2303869743192774_7544220285759651840_n.jpg?_nc_cat=105&_nc_ht=scontent-lax3-1.xx&oh=5c3e9f999103ad8eaf5be7c7df464a46&oe=5D1BACBB |

**1c.**

***Problem Description:*** The system is not casual. In utilizing the step function, we can define vector x to represent input on sample interval over the range of -5 and 9. The output vector y has an interval of -6 to 9. For it to be casualty there can be no output before any other input to a system. To confirm that our system violates the LSI property we subplot the input and output of the system over interval -10 to 10, thus confirming output before input to system.

***Solution: Results:***

|  |  |
| --- | --- |
| clc;  clear all;  close all;  %defines the shift  shift = -1;  z = @(u,w)w.\*u;  %uses the stepseq function and defines interva;  [u1, w1] = stepseq(0,-5,9);  [u2, w2] = sigshift(u1,w1,shift);  %signal add  [z,w] = sigadd(u1,w1,u2,w2);  %plots the graph  subplot(2,1,1);  stem(w1, u1);  title('u(w)');  axis([-10,10, min(z)-1, max(z)+1])  %plots the graph  subplot(2,1,2);  stem(w,z);  title('z(w)');  axis([-10,10, min(z)-1, max(z)+1]) | https://scontent-lax3-1.xx.fbcdn.net/v/t1.15752-9/54220431_290978598246786_8768456621350191104_n.jpg?_nc_cat=109&_nc_ht=scontent-lax3-1.xx&oh=34c470cb0f1f85a43e87f38058b59c96&oe=5D0AC893 |

**Part 5: Constant Coefficient Difference Equations**

**1.**

***Problem Description:*** Determine a function that will comput the output of the casual sytem of FOAR equation. It consist of an input vector x that has a sequence x[n] over the interval 0 to N-1 and an output vector y over the same interval. Starting at n = 1, we utlize a for loop for the indexed values of y[n].

***Solution: Results:***

|  |  |
| --- | --- |
| function y=diffeqn(a,x,yn1)  % Initialize y  y = [];  % for n = 1 to n = the size of x  for n = (1: length(x))  % For the first n we use yn1  if (n == 1)  y = [y, a \* yn1 + x(n)];  else  y = [y, a \* y(n - 1) + x(n)];  end  end  end | No output |

**2.**

***Problem Description:*** Utilizing the FOAR equation to plot each output using stem. We are given to assume a=1 and y[-1]=0. To determine the output over the range 0 to N-1; N=31.

***Solution: Results:***

|  |  |
| --- | --- |
| N = 31;  a = 1;  yn1 = 0;  % First graph (a)  hold on;  subplot(2, 1, 1);  x = [];  % For n = 0 to n = N - 1  for n = (0: N - 1)  % if n is equal to 1, result is 0  % otherwise the result is 1  if (n == 1)  x = [x, 0];  else  x = [x, 1];  end  end  % Draw the graph  stem((0: N-1), diffeqn(a, x, yn1));  % Add a title  title("Using x1[n] = delta[n]");  % Second graph (b)  subplot(2, 1, 2);  x = [];  for n = (0: N - 1)  % Get result from u() which return 1  % if n >= 0  x = [x, u(n)];  end  % Draw the graph  stem((0: N-1), diffeqn(a, x, yn1));  % Add a title  title("Using x1[n] = u[n]")  hold off; |  |

**3.**

***Problem Description:*** Utilizing FOAR equation again, we test it for linearity. With given data a=1 and y[-1]=-1 and N=11, we determine the left hand side as ylhs[n] as FOAR output, and the right hand side.

***Solution: Results:***

|  |  |
| --- | --- |
| clc;  clear all;  close all;  %define variables  c = 1;  z\_v1 = -1;  V = 11;  %define A and B  A = 1;  B = 1;  %defines the u2 and u over the certain interval;  [u1, v1] = stepseq(0,0,V-1);  u2 = 2\*u1;  u = A\*u1 + B\*u2;  %output for the foar input  z\_lhs = diffeqn(c,u,z\_v1);  z1 = diffeqn(c,u1,z\_v1);  z2 = diffeqn(c,u2,z\_v1);  z\_rhs = A\*z1+B\*z2;  %subplots and stem plot for a sequences  figure(3);  subplot(2,1,1);  stem(v1,z\_lhs);  title('zlhs');  xlabel('v');  ylabel('zlhs');  auis([-10,15, 0, 35])  %another sequence  subplot(2,1,2);  stem(v1,z\_rhs);  title('zrhs');  xlabel('v');  ylabel('zrhs');  auis([-10,15, 0, 35]) |  |

**4.**

***Problem Description:*** Test the FOAR casual system of BIBO. We use stem and subplot to plot both outputs.

***Solution: Results:***

|  |  |
| --- | --- |
| clc;  clear all;  close all;  %defines the variables  c = 3/4;  z\_v1 = 0;  V = 31;  %testing by assuming  [u1, v1] = stepseq(0,0,V-1);  z1 = diffeqn(c,u1,z\_v1);  %uses stem and subplot to plot the outputs  figure(4);  subplot(2,1,1);  stem(v1,z1);  %titles the graph  title('z1(v)');  xlabel('v');  ylabel('z1(v)');  axis([-10,35, min(v1)-1, max(v1)+1])  %uses stem and subplot to plot the outputs  z+v1 = -1;  z2 = diffeqn(a,u1,z\_v1);  subplot(2,1,2);  stem(v1,z2);  %titles the graph  title('z2(v)');  xlabel('v');  ylabel('z2(v)');  axis([-10,35, min(z1)-1, max(z1)+1]) |  |

***Encountered Problems***: ***This section required the use of the FOAR equation. Some trouble implementing the stepseq and understanding the subplot.***