ELEC 342 Lab 3: The Discrete Time Fourier Transform and Introduction to Simulink

Name:	JUNPENG GAI
ID number:	40009896
Course number:	ELEC342-X Lab 2224
Date performed:	Monday, 6 March 2023
Due Date:	Monday, 20 March 2023
Lab Instructor Name:	Salameh, Ahmed

"I certify that this submission is my original work and meets the Faculty's Expectations of Originality"

Contents

1.Objectives

In firs section of this lab we are going to practice on discrete time fourier transform (DTFT) on the input signal by using loops to calculate the output signal and draw the magnitude spectrum. In the second part we will get to learn how to use the Simulink within MATLAB, it's a graphical programming environment for the design, visualize and analysis of the design.

2.Theory

Here is how the MATLAB implements the DTFT.

$$X\left(e^{jw}\right) = \sum_{n=-\infty}^{n=\infty} x[n] \times e^{-jw}$$

If n is defined within a certain range **length_n** and w is defined of a certain frequency range **length_w**. Then we will have a nested loop. The outer loop has **length_w** iterations and inner loop has **length_n** iterations.so we will have a **length_w** by **length_n** matrix. When we need to plot the magnitude of the value, we have to use the abs() to calculate the magnitude of the complex number.

Size(variable) will return the size of the variable, if our variable is a matrix of 2 by 3 for example, Size(variable) will return an array with 2 variables [2 3].

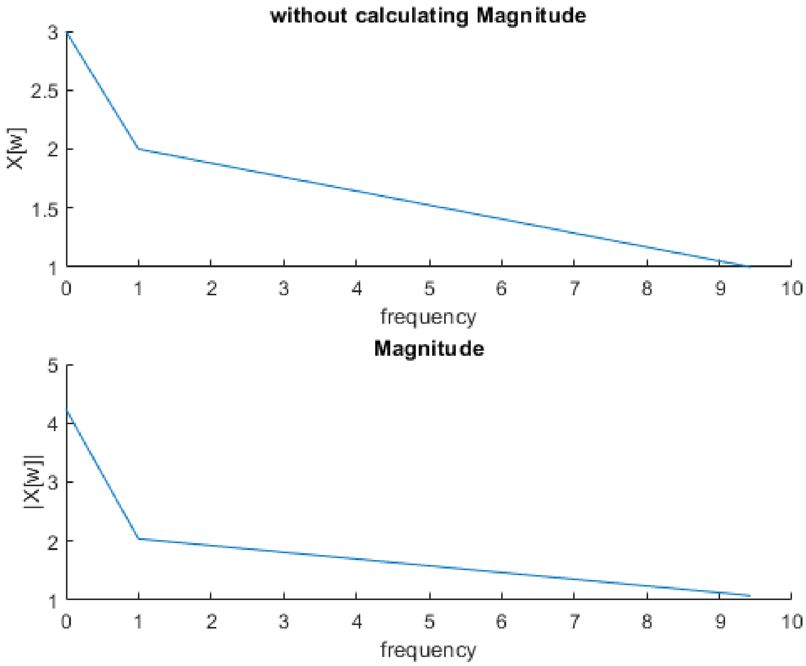
3.Tasks/Results/Discussion

3.1 Complex number

```
%Name:Junpeng gai

%SID:40009896

mycomplex = [ 3 + i*3 2 + i*0.4 1 + i*0.4 ];

w=[0 1 3*pi];

subplot(2,1,1)

hold on

title('without calculating Magnitude')

xlabel('frequency')

ylabel('X[w]')

plot(w,mycomplex);

hold off

mag_of_mycomplex=abs(mycomplex)

subplot(2,1,2)

hold on

title('Magnitude')

xlabel('frequency')

ylabel('|X[w]|')

plot(w,mag_of_mycomplex);

hold off
```

```
Warning: Imaginary parts of complex X and/or Y arguments

ignored.

mag_of_mycomplex =
```

```
    4.2426    2.0396    1.0770
```



We can see that because a complex number has an imaginary part so the magnitude we get is slightly higher than simply plotting the graph. Also there will be a warning if we directly plot the (w,signal).

3.2 size() function

```
%Name:Junpeng gai
%SID:40009896
oneDarray=[1 2 3];
twoDarray=[1 2 3;2 4 2];
sizeofoneD=size(oneDarray)
lengthofoneD=length(oneDarray)


sizeoftwoD=size(twoDarray)
lengthoftwoD=length(twoDarray)
```

```
sizeofoneD =

     1     3

lengthofoneD =

     3

sizeoftwoD =

     2     3

lengthoftwoD =

     3
```

We can see that the size will return the dimension of the matrix and the length of the matrix in one single array, but **sizeoftwoD**=size(twoDarray,2) will give us the number of columns and **sizeoftwoD**=size(twoDarray,1) will give us the number of rows.

3.3 Simulink

We create 2 blocks to observe the sin signal with a scope block and set the frequency to $\frac{2\pi}{150}$ and amplitude to 1.

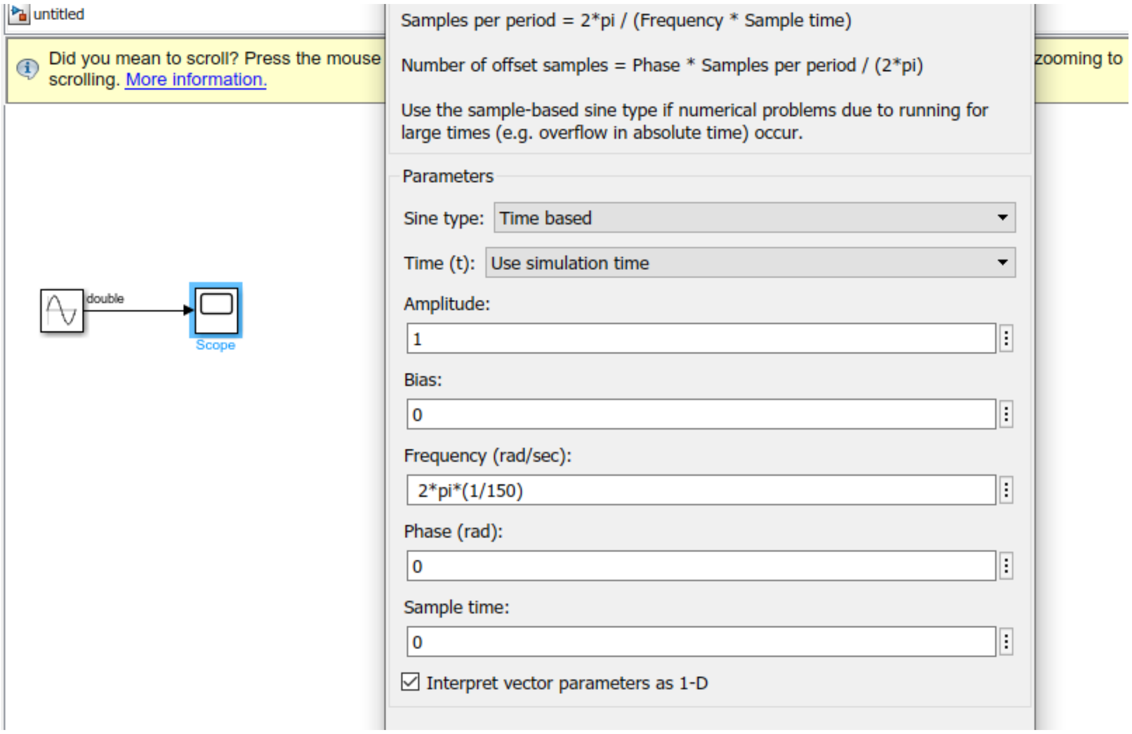


Figure 1 parameters for sine wave source block

After that we set the parameter for the scope block

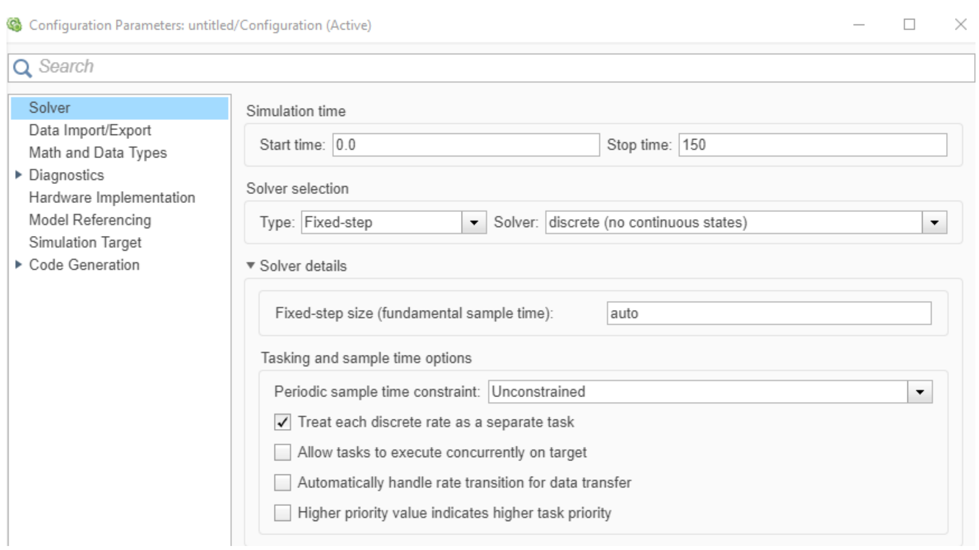


Figure 2 Scope block configuration

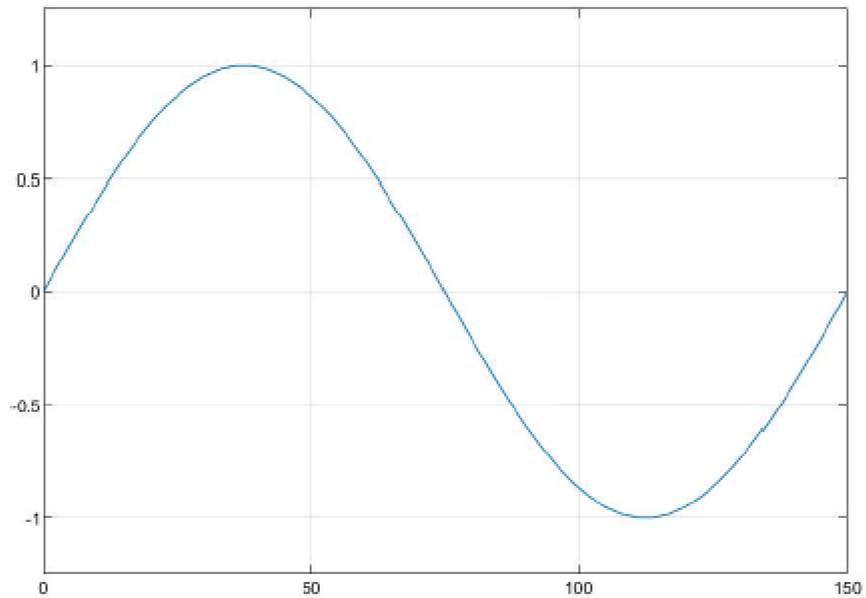


Figure 3 Here is the signal we observed from the scope

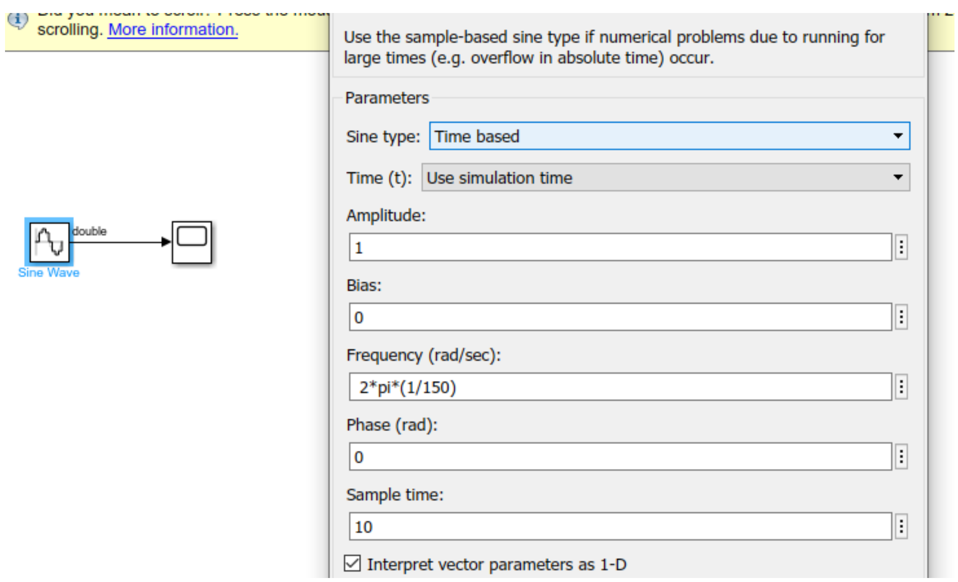


Figure 4 set the sample time to 10

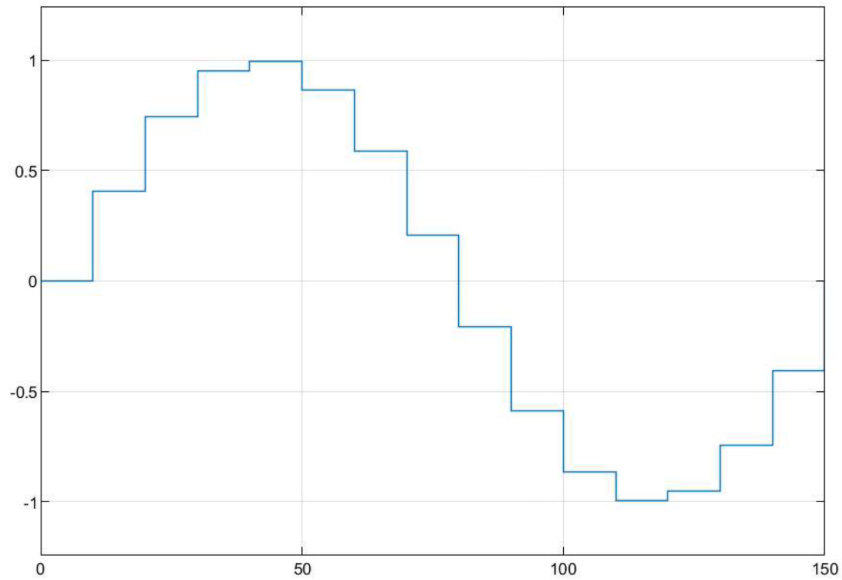


Figure 5 Graph when time step is 10

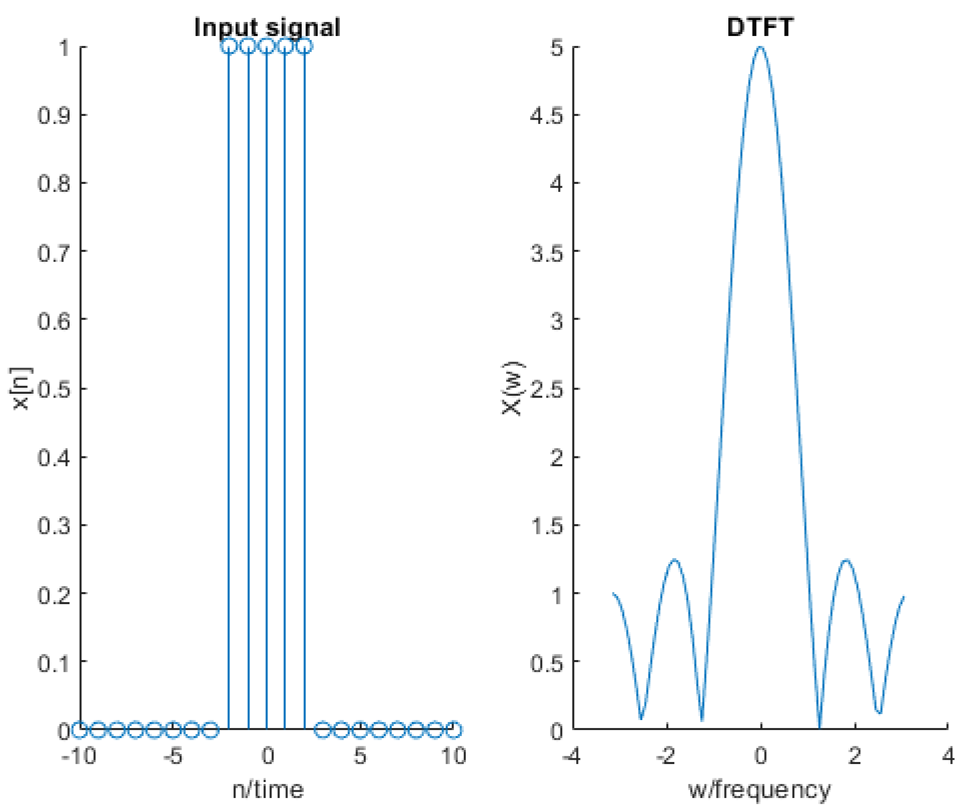
4.Questions

4.1 Question 1

Obtain the DTFT of a pulse x[n] shown in Figure 2 over the interval -10 <= n <= 10. Compute the transform over the frequency interval -pi <= w <= pi using a user-defined input value for the step size. Plot the signal x[n] (using stem) together with its transform (using plot). Use the length function to control the number of iterations of the outer for loop given in the pseudocode algorithm.

```
x=zeros([1 21]);

x(9)=1;

x(10)=1;

x(11)=1;

x(12)=1;

x(13)=1;

n=-10:10;

stem(n,x);

w=-pi:0.1:pi;

length_w=length(w);



length_n=length(n);

sum=zeros([1 length_w]);

subplot(1,2,1)

hold on
stem(n,x)
title('Input signal')
xlabel('n/time')
ylabel('x[n]')
hold off
```

```
    for frequency = 1:length_w

        for index = 1:length_n

        sum(frequency) = sum(frequency) + (x(index) * exp((-j*w(frequency)*n(index ))));

        end
    end

    subplot(1,2,2)
    hold on
    title('DTFT')
    xlabel('w/frequency')
    ylabel('X(w)')
    sum=abs(sum);
    plot(w,sum)
    hold off
```

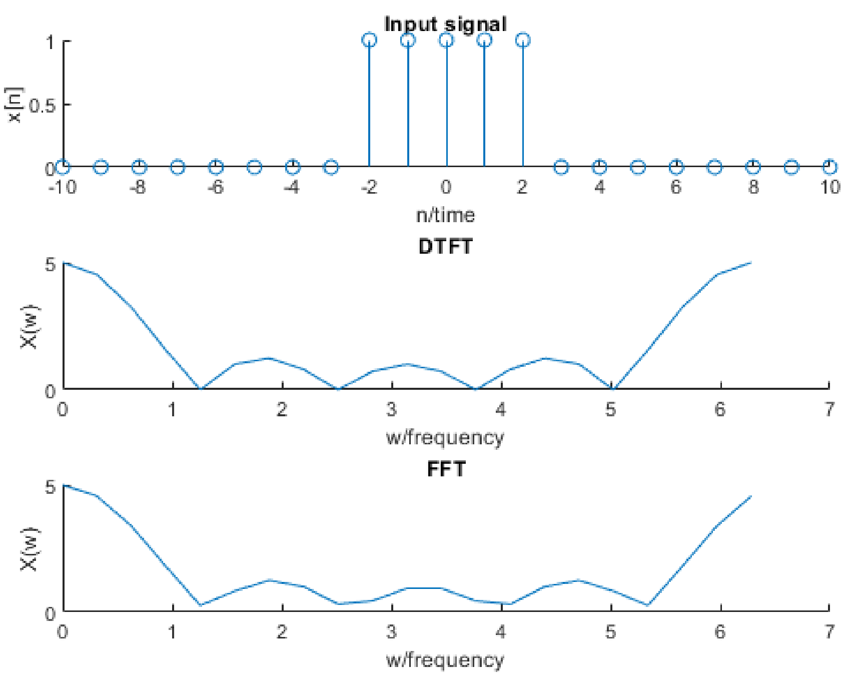


Published with MATLAB® R2020a

4.2 Question 2

```
%Name:Junpeng gai

%SID:40009896

x=zeros([1 21]);

x(9)=1;

x(10)=1;

x(11)=1;

x(12)=1;

x(13)=1;

n=-10:10;

stem(n,x);

w=0:pi/10:2*pi;

length_w=length(w);


length_n=length(n);

sum=zeros([1 length_w]);

subplot(3,1,1)

hold on

stem(n,x)

title('Input signal')

xlabel('n/time')

ylabel('x[n]')

hold off

for frequency = 1:length_w


     for index = 1:length_n


     sum(frequency) = sum(frequency) + (x(index) * exp((-j*w(frequency)*n(index ))));


     end
```

```
end


subplot(3,1,2)

hold on

title('DTFT')

xlabel('w/frequency')

ylabel('X(w)')

sum=abs(sum);

plot(w,sum)

hold off

x_fft = fft(x);

subplot(3,1,3)

hold on

title('FFT')

xlabel('w/frequency')

ylabel('X(w)')

mag_x_fft=abs(x_fft);

plot(w,mag_x_fft)

hold off
```

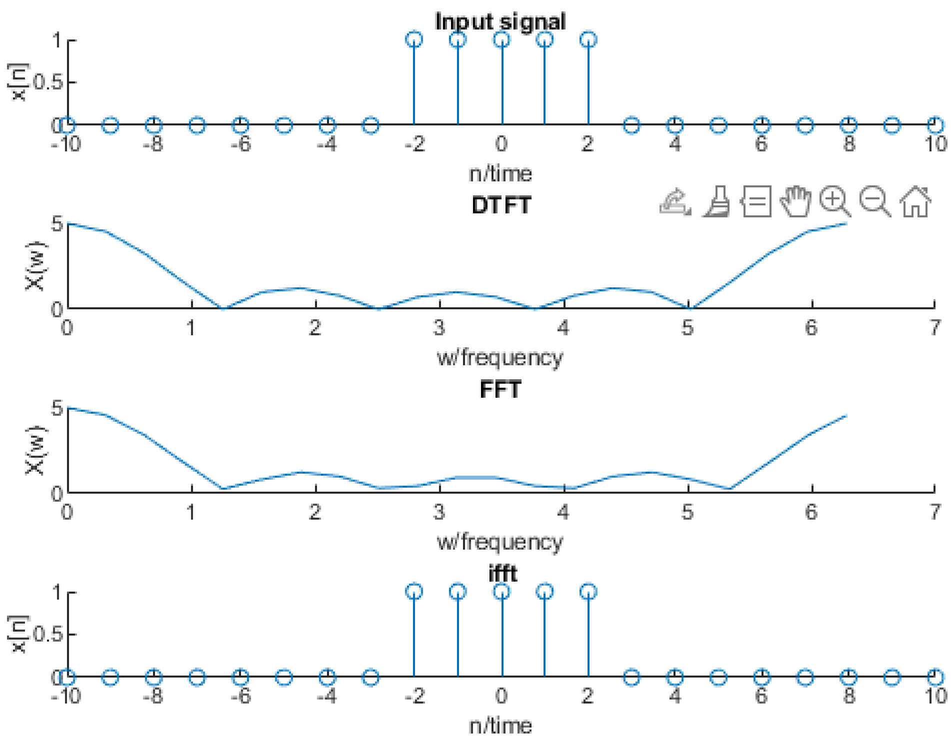


Published with MATLAB® R2020a

4.3 Question 3

```
%Name:Junpeng gai
%SID:40009896
x=zeros([1 21]);
x(9)=1;
x(10)=1;
x(11)=1;
x(12)=1;
x(13)=1;
n=-10:10;
stem(n,x);
w=0:pi/10:2*pi;
length_w=length(w);

length_n=length(n);
sum=zeros([1 length_w]);
subplot(4,1,1)
hold on
stem(n,x)
title('Input signal')
xlabel('n/time')
ylabel('x[n]')
hold off
for frequency = 1:length_w

     for index = 1:length_n

     sum(frequency) = sum(frequency) + (x(index) * exp((-j*w(frequency)*n(index ))));
```

```
    end
end

subplot(4,1,2)
hold on
title('DTFT')
xlabel('w/frequency')
ylabel('X(w)')
sum=abs(sum);
plot(w,sum)
hold off
x_fft = fft(x);
subplot(4,1,3)
hold on
title('FFT')
xlabel('w/frequency')
ylabel('X(w)')
mag_x_fft=abs(x_fft);
plot(w,mag_x_fft)
hold off

subplot(4,1,4)
hold on
title('ifft')
xlabel('n/time')
ylabel('x[n]')
x_original=ifft(x_fft);
stem(n,x_original)
hold off
```



4.4 Part2

I followed the instructions to build the following blocks.

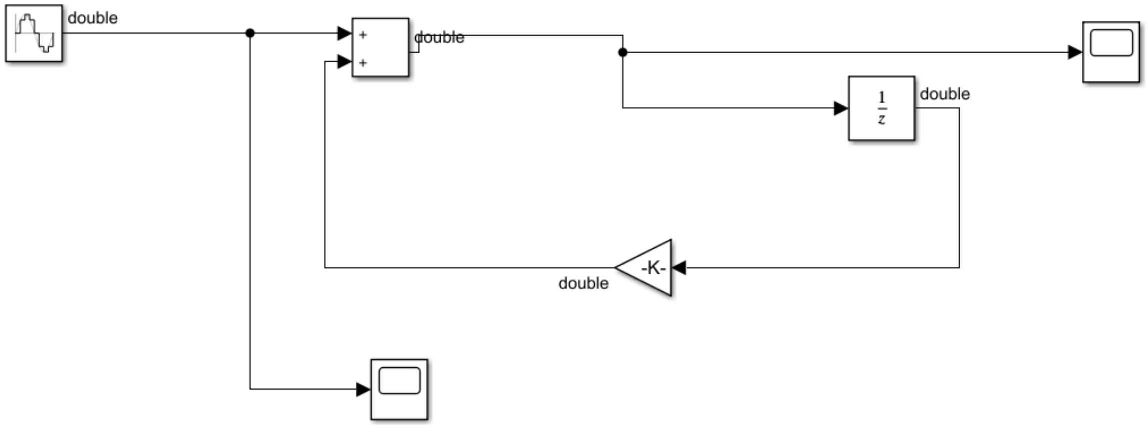


Figure 6 Block models

There are 2 scopes, one on the right is used to observe the output signal, the other one at bottom is used to observe the input signal.

We choose the same frequency as above which is 2pi/150 and in the sample step we set it to 1.

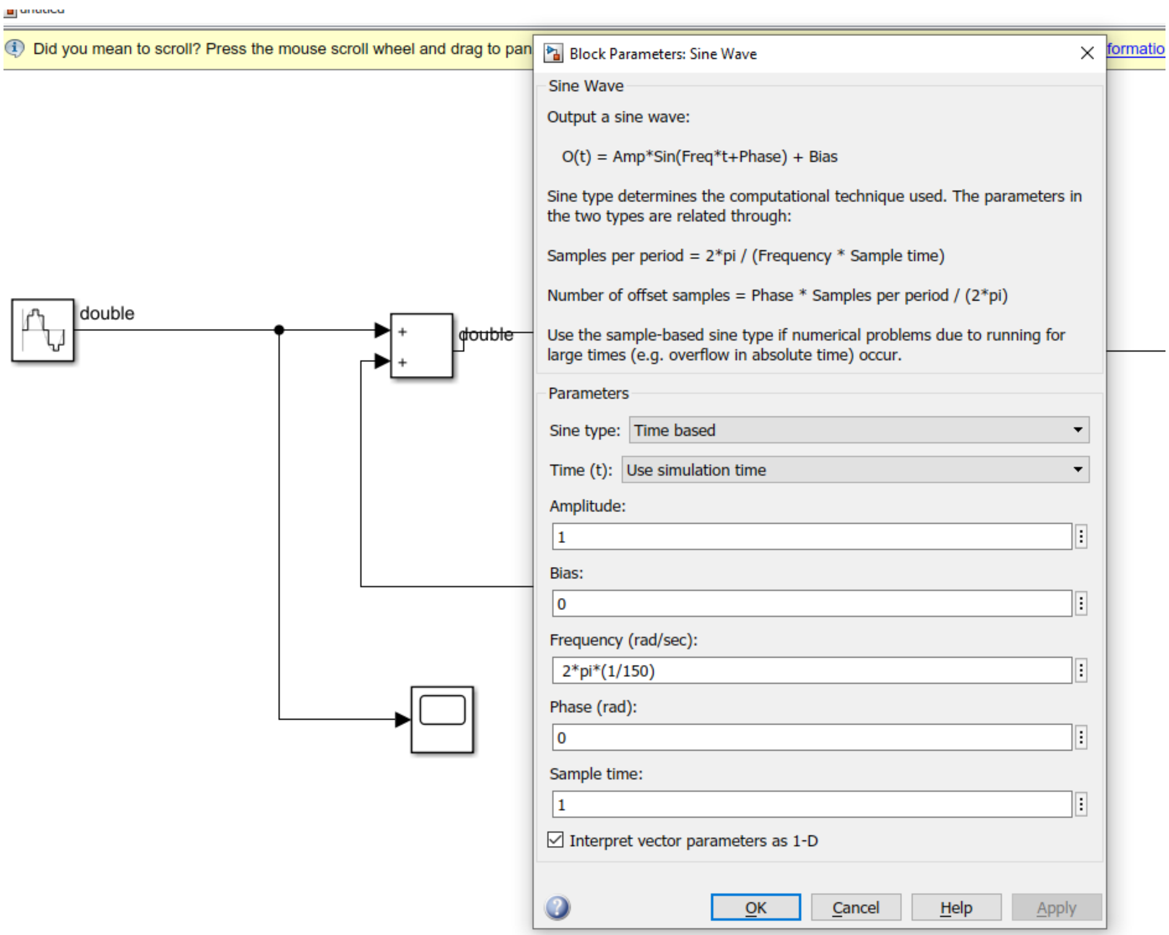


Figure 7 Here is the sine wave source setting

Here is the setting for the output scopes, the same setting will be implemented to both scopes.

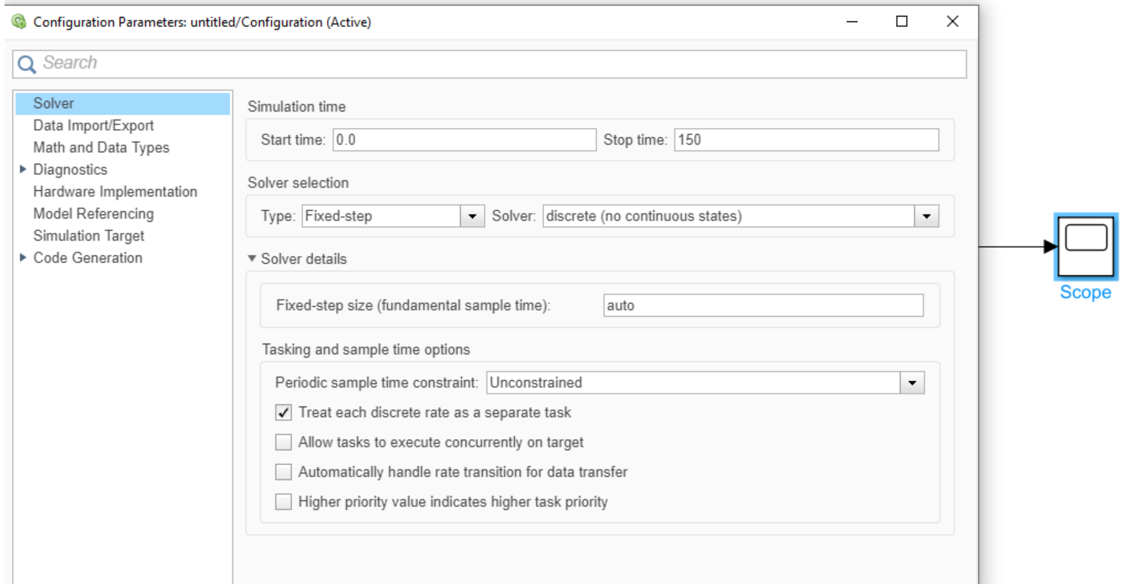


Figure 8Scope block setting

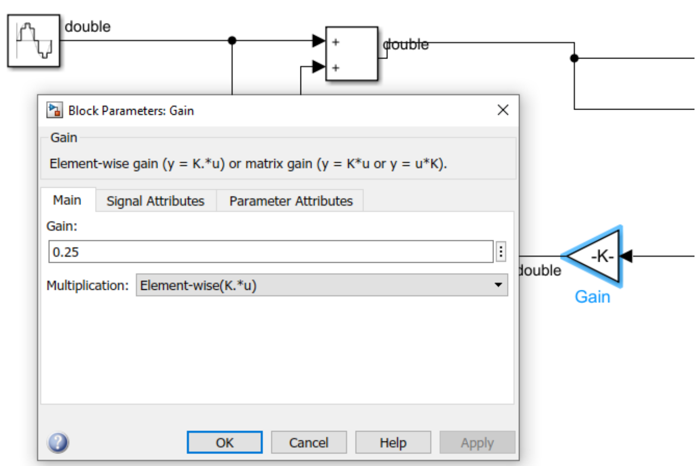


Figure 9 Gain block setting

We run the simulation and here are the outputs from the 2 scopes. The first one is the input x[k] and the second one is the output y[k].

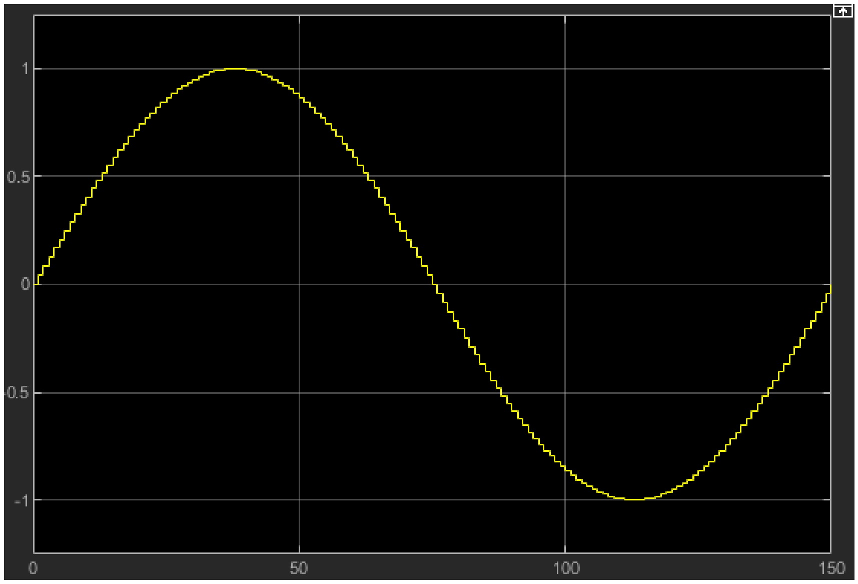


Figure 10 x[k] output from the source

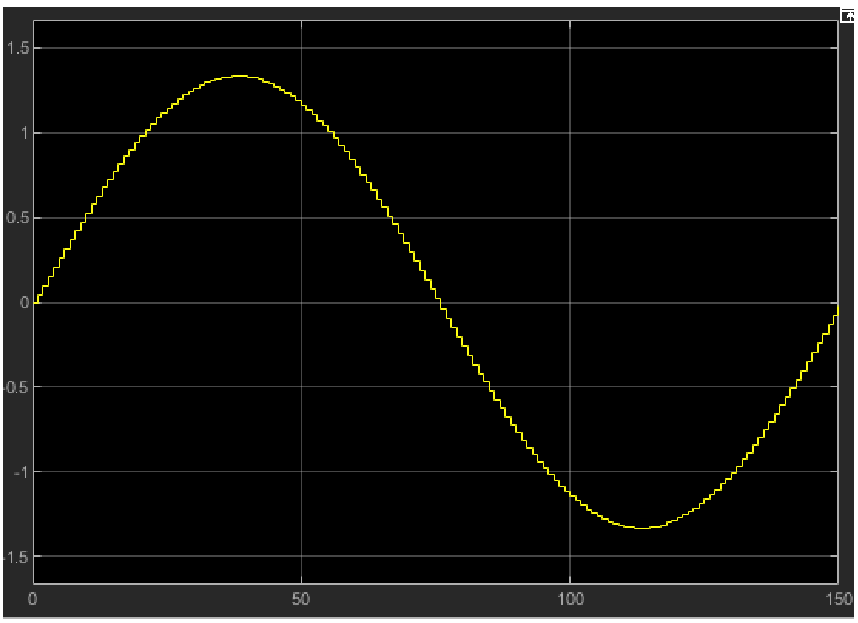
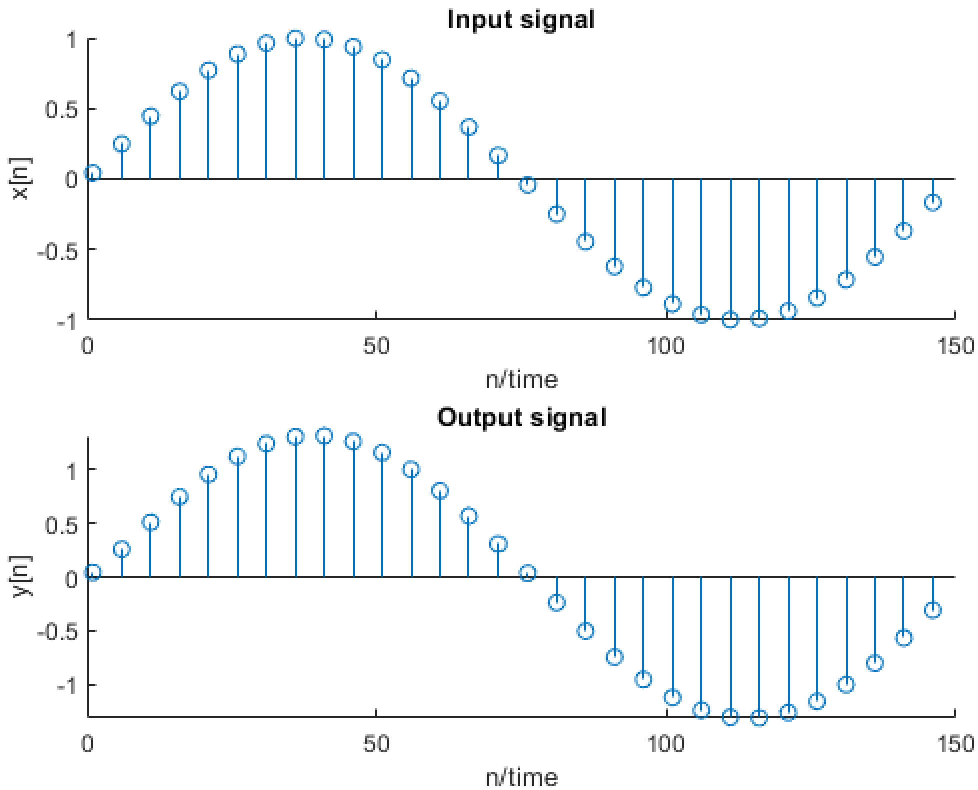


Figure 11 y[k]   output from the scope on the right

In order to make sure of the results, I did the simulation in the MATLAB to double check the result.

```
%Name:Junpeng Gai
%SID:40009896
```

```
n=1:5:150;
x=sin((n*2*pi)/150);
y=zeros(1,30);
y(1)=x(1);
H=zeros(1,30);
for i =2:30
   y(i)=x(i)+0.25*((y(i-1)));
end

for i =1:5:150
   H(i)=0.25^i;
end
subplot(2,1,1)
hold on
stem(n,x);
title('Input signal')
xlabel('n/time')
ylabel('x[n]')
hold off

subplot(2,1,2)
hold on
stem(n,y);
title('Output signal')
xlabel('n/time')
ylabel('y[n]')
hold off
```



Published with MATLAB® R2020a

We can see that our results are the same

5.Conclusions

6.Appendix

6.1 Question 1

```
%Name:Junpeng gai
%SID:40009896
x=zeros([1 21]);
x(9)=1;
x(10)=1;
x(11)=1;
x(12)=1;
x(13)=1;
n=-10:10;
stem(n,x);
w=-pi:0.1:pi;
length_w=length(w);

length_n=length(n);
sum=zeros([1 length_w]);
subplot(1,2,1)
hold on
stem(n,x)
title('Input signal')
xlabel('n/time')
ylabel('x[n]')
hold off
for frequency = 1:length_w

     for index = 1:length_n

     sum(frequency) = sum(frequency) + (x(index) * exp((-j*w(frequency)*n(index ))));

     end
end

subplot(1,2,2)
hold on
title('DTFT')
xlabel('w/frequency')
ylabel('X(w)')
sum=abs(sum);
plot(w,sum)
hold off
```

6.2 Question 2

```
%Name:Junpeng gai
%SID:40009896
x=zeros([1 21]);
x(9)=1;
```

```
x(10)=1;
x(11)=1;
x(12)=1;
x(13)=1;
n=-10:10;
stem(n,x);
w=0:pi/10:2*pi;
length_w=length(w);

length_n=length(n);
sum=zeros([1 length_w]);
subplot(3,1,1)
hold on
stem(n,x)
title('Input signal')
xlabel('n/time')
ylabel('x[n]')
hold off
for frequency = 1:length_w

     for index = 1:length_n

     sum(frequency) = sum(frequency) + (x(index) * exp((-j*w(frequency)*n(index ))));

     end
end

subplot(3,1,2)
hold on
title('DTFT')
xlabel('w/frequency')
ylabel('X(w)')
sum=abs(sum);
plot(w,sum)
hold off
x_fft = fft(x);
subplot(3,1,3)
hold on
title('FFT')
xlabel('w/frequency')
ylabel('X(w)')
mag_x_fft=abs(x_fft);
plot(w,mag_x_fft)
hold off
```

6.3 Question 3

```
%Name:Junpeng gai
%SID:40009896
x=zeros([1 21]);
x(9)=1;
x(10)=1;
x(11)=1;
x(12)=1;
x(13)=1;
n=-10:10;
stem(n,x);
w=0:pi/10:2*pi;
length_w=length(w);

length_n=length(n);
sum=zeros([1 length_w]);
```

```
subplot(3,1,1)
hold on
stem(n,x)
title('Input signal')
xlabel('n/time')
ylabel('x[n]')
hold off
for frequency = 1:length_w

     for index = 1:length_n

     sum(frequency) = sum(frequency) + (x(index) * exp((-j*w(frequency)*n(index ))));

     end
end

subplot(3,1,2)
hold on
title('DTFT')
xlabel('w/frequency')
ylabel('X(w)')
sum=abs(sum);
plot(w,sum)
hold off
x_fft = fft(x);
subplot(3,1,3)
hold on
title('FFT')
xlabel('w/frequency')
ylabel('X(w)')
mag_x_fft=abs(x_fft);
plot(w,mag_x_fft)
hold off
```

6.4 Part 2

```
%Name:Junpeng Gai
%SID:40009896
n=1:1:150;
x=sin((n*2*pi)/150);
y=zeros(1,30);
y(1)=x(1);
H=zeros(1,30);
for i =2:150
   y(i)=x(i)+0.25*((y(i-1)));
end

for i =1:150
   H(i)=0.25^i;
end
subplot(2,1,1)
hold on
stem(n,x);
title('Input signal')
xlabel('n/time')
ylabel('x[n]')
hold off

subplot(2,1,2)
hold on
stem(n,y);
title('Output signal')
xlabel('n/time')
```

```
ylabel('y[n]')
hold off
```