Published July 16, 2014 by Mathuranathan

How to plot FFT using Matlab – FFT of basic signals : Sine and Cosine waves

(**58** votes, average: **4.50** out of 5)

Articles in this series:

Introduction

Numerous texts are available to explain the basics of Discrete Fourier Transform and its very efficient implementation – Fast Fourier Transform (FFT). Often we are confronted with the need to generate simple, standard signals (sine, cosine, Gaussian pulse, squarewave, isolated rectangular pulse, exponential decay, chirp signal) for simulation purpose. I intend to show (in a series of articles) how these basic signals can be generated in Matlab and how to represent them in frequency domain using FFT.

For more such examples check this ebook : Simulation of Digital Communications using Matlab – by Mathuranathan Viswanathan

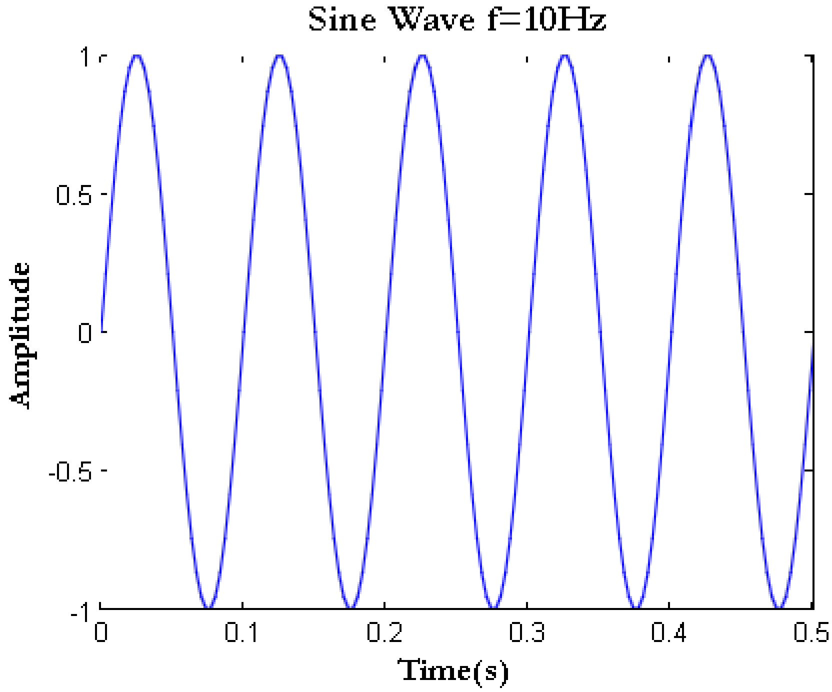
Sine Wave

In order to generate a sine wave in Matlab, the first step is to fix the frequency f of the sine wave. For example, I intend to generate a $f = 10Hz$ sine wave whose minimum and maximum amplitudes are $-1V$ and $+1V$ respectively. Now that you have determined the frequency of the sinewave, the next step is to determine the sampling rate. Matlab is a software that processes everything in digital. In order to generate/plot a smooth sine wave, the sampling rate must be far higher than the prescribed minimum required sampling rate which is at least twice the frequency f – as per Nyquist Shannon Theorem. A oversampling factor of 30 is chosen here – this is to plot a smooth continuous-like sine wave (If this is not the requirement, reduce the oversampling factor to desired level). Thus the sampling rate becomes $f_s = 30f = 300Hz$. If a phase shift is desired for the sine wave, specify it too.

```
f=10; %frequency of sine wave
overSampRate=30; %oversampling rate
fs=overSampRate*f; %sampling frequency
phase = 1/3*pi; %desired phase shift in radians
nCyl = 5; %to generate five cycles of sine wave

t=0:1/fs:nCyl*1/f; %time base

x=sin(2*pi*f*t+phase); %replace with cos if a cosine wave is desired
plot(t,x);
title(['Sine Wave f=', num2str(f), 'Hz']);
xlabel('Time(s)');
ylabel('Amplitude');
```



Representing in Frequency Domain

Representing the given signal in frequency domain is done via Fast Fourier Transform (FFT) which implements Discrete Fourier Transform (DFT) in an efficient manner. Usually, power spectrum is desired for analysis in frequency domain. In a power spectrum, power of each frequency component of the given signal is plotted against their respective frequency. The command FFT(x,N) computes the N-point DFT. The number of points – N – in the DFT computation is taken as power of 2 for facilitating efficient computation with FFT. A value of $N = 1024$ is chosen here. It can also be chosen as next power of 2 of the length of the signal.

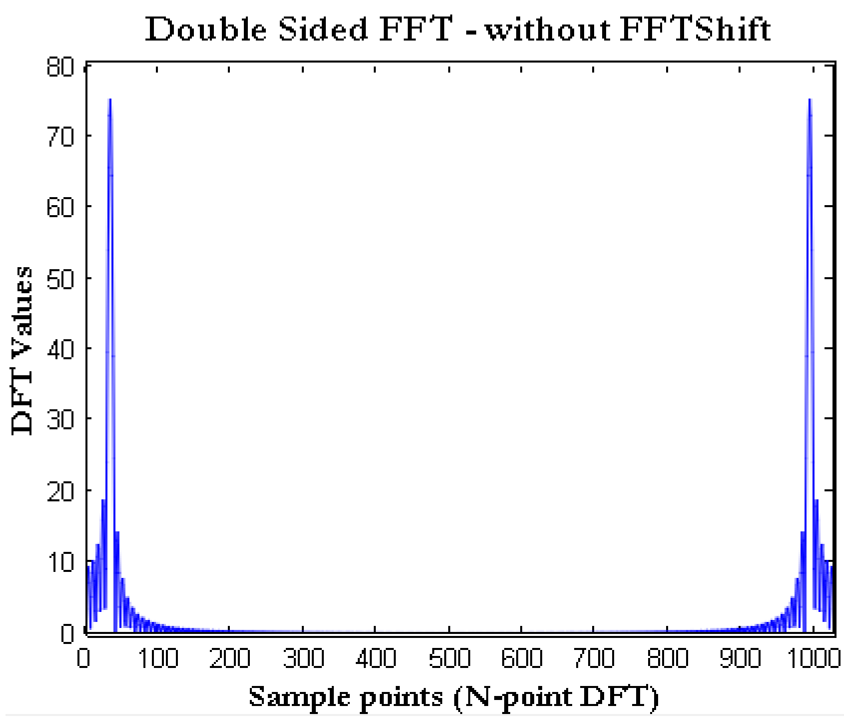
Different representations of FFT:

Since FFT is just a numeric computation of N-point DFT, there are many ways to plot the result.

1. Plotting raw values of DFT:

The x-axis runs from 0 to $N-1$ – representing N sample values. Since the DFT values are complex, the magnitude of the DFT ($abs(X)$) is plotted on the y-axis. From this plot we cannot identify the frequency of the sinusoid that was generated.

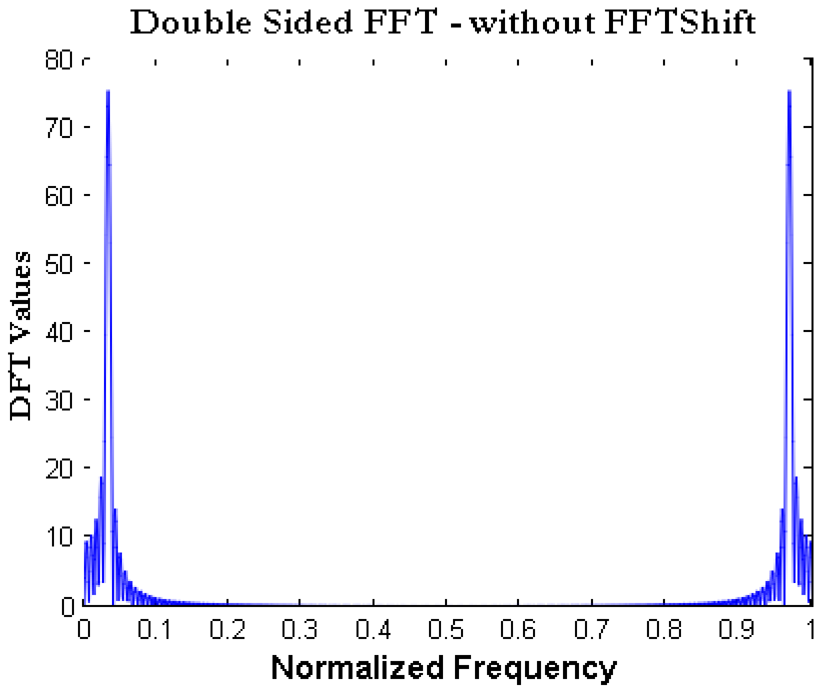
```
NFFT=1024; %NFFT-point DFT
X=fft(x,NFFT); %compute DFT using FFT
nVals=0:NFFT-1; %DFT Sample points
plot(nVals,abs(X));
title('Double Sided FFT - without FFTShift');
xlabel('Sample points (N-point DFT)')
ylabel('DFT Values');
```



2. FFT plot – plotting raw values against Normalized Frequency axis:

In the next version of plot, the frequency axis (x-axis) is normalized to unity. Just divide the sample index on the x-axis by the length N of the FFT. This normalizes the x-axis with respect to the sampling rate f_s. Still, we cannot figure out the frequency of the sinusoid from the plot.

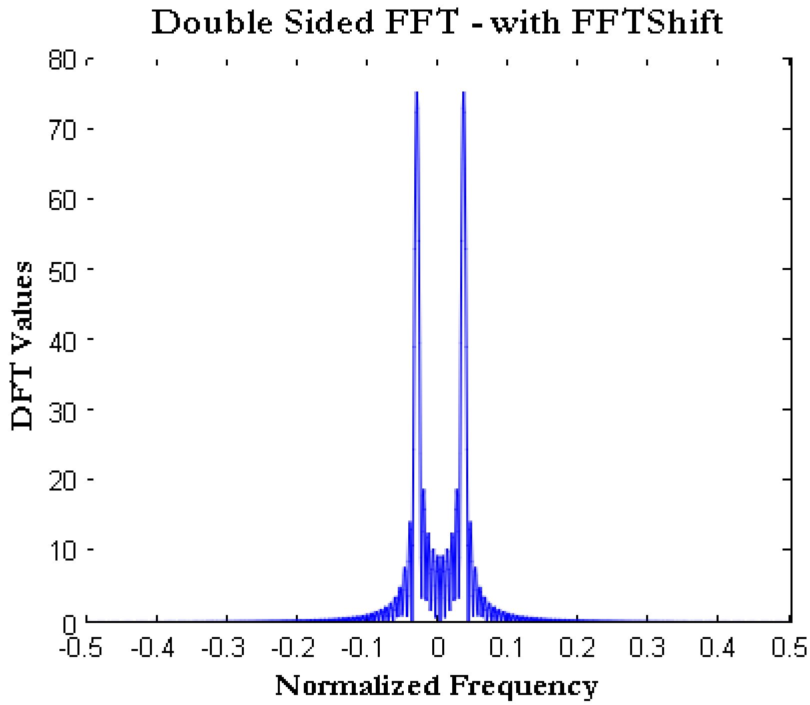
```
NFFT=1024; %NFFT-point DFT
X=fft(x,NFFT); %compute DFT using FFT
nVals=(0:NFFT-1)/NFFT; %Normalized DFT Sample points
plot(nVals,abs(X));
title('Double Sided FFT - without FFTShift');
xlabel('Normalized Frequency')
ylabel('DFT Values');
```



3. FFT plot – plotting raw values against normalized frequency (positive & negative frequencies):

As you know, in the frequency domain, the values take up both positive and negative frequency axis. In order to plot the DFT values on a frequency axis with both positive and negative values, the DFT value at sample index 0 has to be centered at the middle of the array. This is done by using $FFTshift$ function in Matlab. The x-axis runs from -0.5 to 0.5 where the end points are the normalized 'folding frequencies' with respect to the sampling rate f_s.

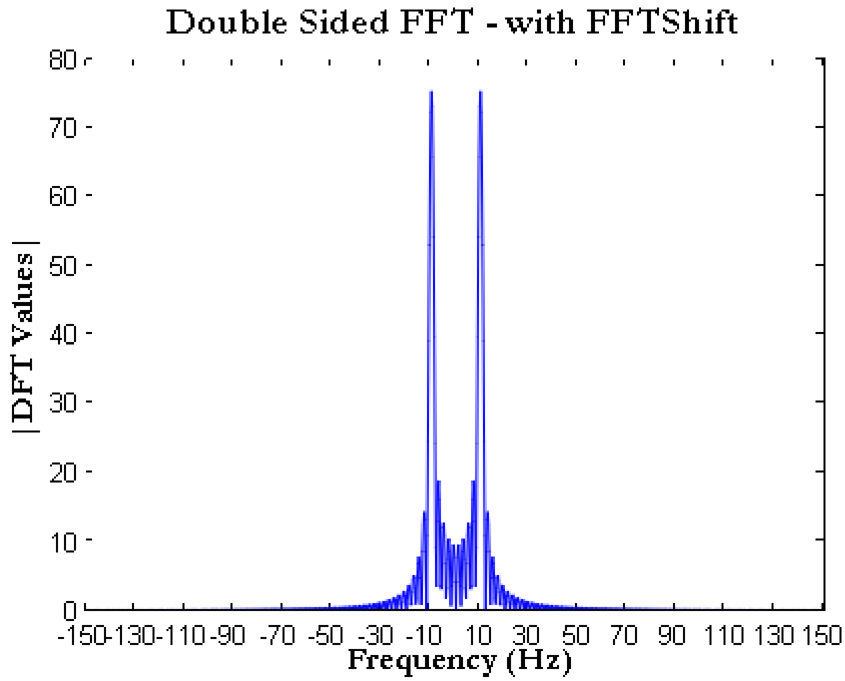
```
NFFT=1024; %NFFT-point DFT
X=fftshift(fft(x,NFFT)); %compute DFT using FFT
fVals=(-NFFT/2:NFFT/2-1)/NFFT; %DFT Sample points
plot(fVals,abs(X));
title('Double Sided FFT - with FFTShift');
xlabel('Normalized Frequency')
ylabel('DFT Values');
```



4. FFT plot – Absolute frequency on the x-axis Vs Magnitude on Y-axis:

Here, the normalized frequency axis is just multiplied by the sampling rate. From the plot below we can ascertain that the absolute value of FFT peaks at $10Hz$ and $-10Hz$. Thus the frequency of the generated sinusoid is $10Hz$. The small side-lobes next to the peak values at $10Hz$ and $-10Hz$ are due to spectral leakage.

```
NFFT=1024;
X=fftshift(fft(x,NFFT));
fVals=fs*(-NFFT/2:NFFT/2-1)/NFFT;
plot(fVals,abs(X),'b');
title('Double Sided FFT - with FFTShift');
xlabel('Frequency (Hz)')
ylabel('|DFT Values|');
```



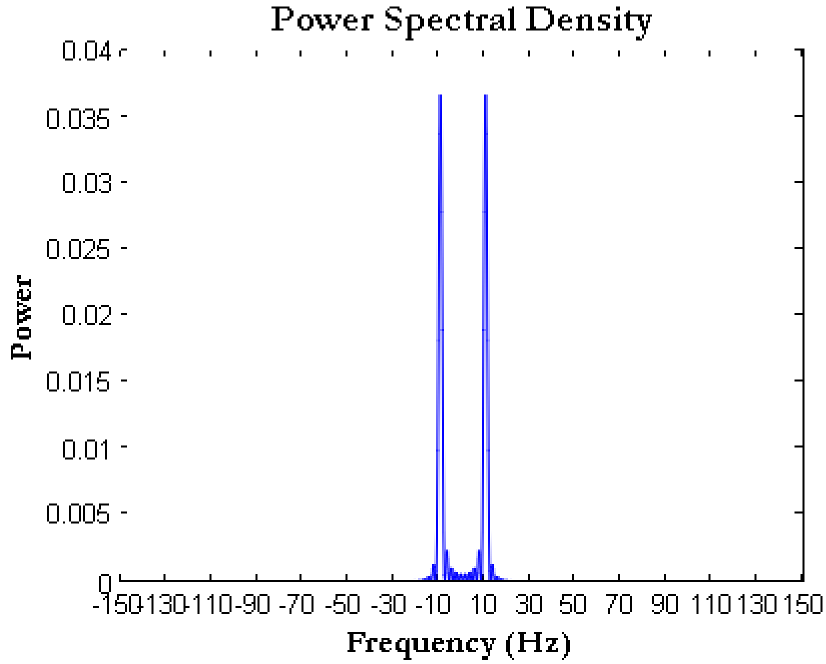
5. Power Spectrum – Absolute frequency on the x-axis Vs Power on Y-axis:

The following is the most important representation of FFT. It plots the power of each frequency component on the y-axis and the frequency on the x-axis. The power can be plotted in linear scale or in log scale. The power of each frequency component is calculated as

$$P_x(f) = X(f)X^*(f)$$

Where $X(f)$ is the frequency domain representation of the signal $x(t)$. In Matlab, the power has to be calculated with proper scaling terms (since the length of the signal and transform length of FFT may differ from case to case).

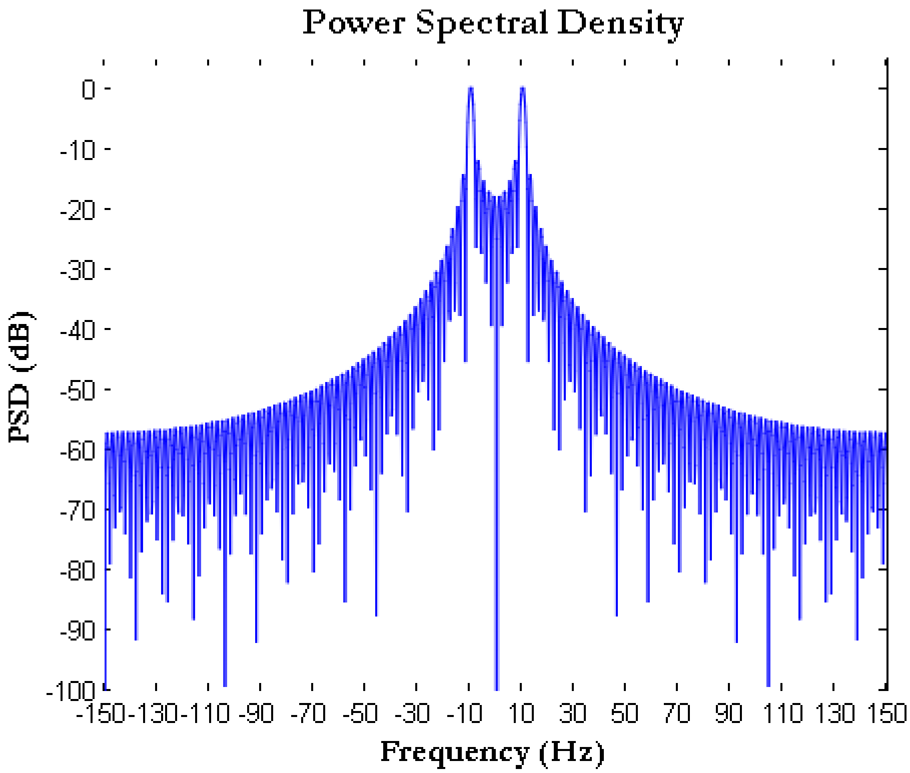
```
NFFT=1024;
L=length(x);
X=fftshift(fft(x,NFFT));
Px=X.*conj(X)/(NFFT*L); %Power of each freq components
fVals=fs*(-NFFT/2:NFFT/2-1)/NFFT;
plot(fVals,Px,'b');
title('Power Spectral Density');
xlabel('Frequency (Hz)')
ylabel('Power');
```



[If you wish to verify the total power of the signal from time domain and frequency domain plots, follow this link.]

Plotting the PSD plot with y-axis on log scale produces the most encountered type of PSD plot in signal processing.

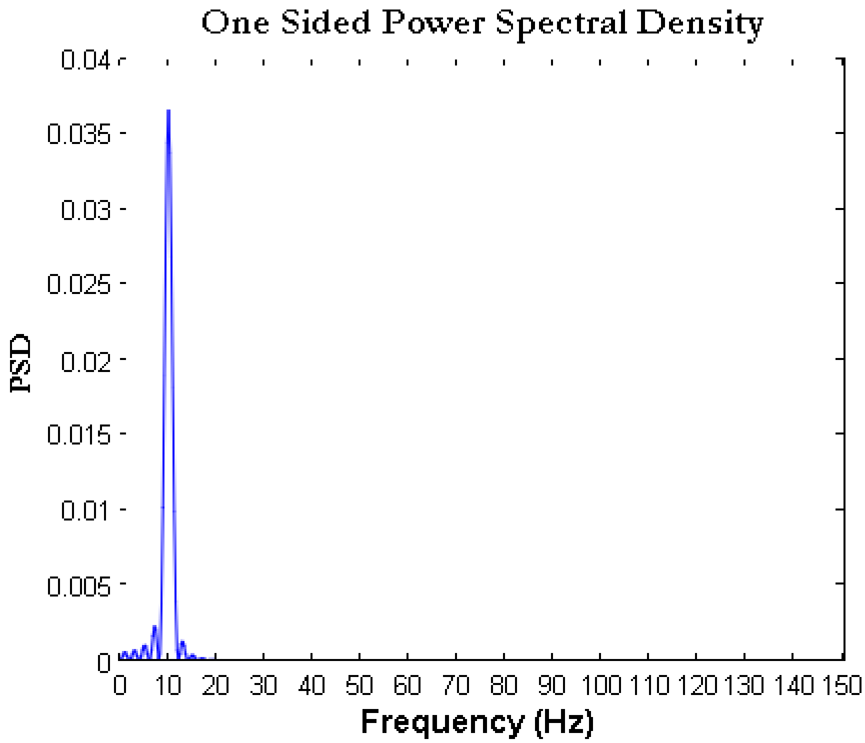
```
NFFT=1024;
L=length(x);
X=fftshift(fft(x,NFFT));
Px=X.*conj(X)/(NFFT*L); %Power of each freq components
fVals=fs*(-NFFT/2:NFFT/2-1)/NFFT;
plot(fVals,10*log10(Px),'b');
title('Power Spectral Density');
xlabel('Frequency (Hz)')
ylabel('Power');
```



6. Power Spectrum – One-Sided frequencies

In this type of plot, the negative frequency part of x-axis is omitted. Only the FFT values corresponding to 0 to $N/2)$ sample points of N-point DFT are plotted. Correspondingly, the normalized frequency axis runs between 0 to 0.5. The absolute frequency (x-axis) runs from 0 to $f_s/2$.

```
L=length(x);
NFFT=1024;
X=fft(x,NFFT);
Px=X.*conj(X)/(NFFT*L); %Power of each freq components
fVals=fs*(0:NFFT/2-1)/NFFT;
plot(fVals,Px(1:NFFT/2),'b','LineSmoothing','on','LineWidth',1);
title('One Sided Power Spectral Density');
xlabel('Frequency (Hz)')
ylabel('PSD');
```



Rate this article: **(58 votes, average: 4.50 out of 5)**

Articles in this series:
How to Interpret FFT results – complex DFT, frequency bins and FFTShift
How to Interpret FFT results – obtaining Magnitude and Phase information
FFT and Spectral Leakage
How to plot FFT using Matlab – FFT of basic signals : Sine and Cosine waves (this article)
Generating Basic signals – Square Wave and Power Spectral Density using FFT
Generating Basic signals – Rectangular Pulse and Power Spectral Density using FFT
Generating Basic Signals – Gaussian Pulse and Power Spectral Density using FFT
Chirp Signal – Frequency Sweeping – FFT and power spectral density
Constructing the Auto Correlation Matrix using FFT

For more such examples check this ebook : Simulation of Digital Communications using Matlab – by Mathuranathan Viswanathan

Recommended Signal Processing Books

Share this:

Previous Post

Generating Multiple Sequences of correlated random variables

Next Post

Generating Basic signals – Square Wave and Power Spectral Density using FFT

Mathuranathan

Mathuranathan Viswanathan - Founder and Author @ gaussianwaves.com which has garnered worldwide readership. He is a masters in communication engineering and has 9 years of technical expertise in channel modeling and has worked in various technologies ranging from read channel design for hard drives, GSM/EDGE/GPRS, OFDM, MIMO, 3GPP PHY layer and DSL. He also specializes in tutoring on various subjects like signal processing, random process, digital communication etc.., LinkedIn Profile

LATEST ARTICLES MATLAB CODES SIGNAL PROCESSING TIPS & TRICKS

FFT FOURIER ANALYSIS FOURIER TRANSFORM MATLAB CODE POWER SPECTRAL DENSITY PSD TIPS & TRICKS

7 Comments **Gaussianwaves** 1 **Login**

Recommend **Share** **Sort by Best**

Join the discussion…

Mohand • 9 months ago

Hi everybody,

I need to do fft to obtain the frequency content of my signal and calculate the energy to obtain the transmission coefficient, but I have a lot of problems since I don't know exatly what is the correct way to obtain everything with a correct scale and units.

I have an output signal (voltage vs time : size(u)=(100000,2)) obtained from excitation:

pulse (pulse width = 100 µs); on the oscilloscope : duration is 10 ms and 10 MSa/s of sampling rate.

for the frequencies, I expect to get a fundamental at around 8 KHz and the harmonics (2nd and third).

this is a very smal part of my signal :
t(s) t(volts)
3.99000000E-02 1.3574E-02
3.99001000E-02 1.4299E-02
3.99002000E-02 1.3969E-02
3.99003000E-02 1.3508E-02

see more

Reply • Share ›

Yamuna • a year ago

Hi Sir,

In the above example when I gave the number of cycles as 10, then it is giving a spectrum which has a higher amplitude. This seems so strange to me because the number of cycles is just a visualisation parameter and that should not chance the appearance of the spectrum.

nCyl = 10; %to generate ten cycles of sine wave

Kindly explain why this is happening.

Reply • Share ›

Mathuranathan Mod → Yamuna • a year ago

I was plotting the NON-normalized magnitude spectrum. The amplitude is influenced by the number of samples taken to compute DFT. For normalized magnitude spectrum please check the following post.
http://www.gaussianwaves.co...

Reply • Share ›

ArrozConCostra • 2 years ago

Thanks very much, very explanatory article

There's an error on point 3. It should read
X=fftshift(x,NFFT); %compute DFT using FFT

Rgds

Reply • Share ›

Mathuranathan Mod → ArrozConCostra • a year ago

Thanks for pointing it out

Reply • Share ›

Reply • Share ›

Antonis Tsiflikiotis • 2 years ago

Hi!
Can you plean explain me why in Psd:
Px=X.*conj(X)/(NFFT*L)

you divide by (NFFT*L) and not L^2?

Reply • Share ›

ArrozConCostra → Antonis Tsiflikiotis • 2 years ago

I would ask why he does not divide only by L, as this is the usual scaling factor when calculating FFT

Reply • Share ›

ALSO ON **GAUSSIANWAVES**

Introduction to Multiple Antenna Systems

9 comments • 3 years ago•

mikel — Thank you very much for your fast reply!

Sampling Theorem – Bandpass or Intermediate or Under Sampling

10 comments • 3 years ago•

ali hassan — dear Mr Mathuranathan I want you to help in my wife project how can contact you please it is so urgent .

Generating Basic signals – Square Wave and Power Spectral Density using

5 comments • 3 years ago•

Mathuranathan — Dharman JeyaramanSome books listed below1) MATLAB/Simulink for Digital

Ergodic Capacity of a SISO system over a Rayleigh Fading channel – Simulation

6 comments • 2 years ago•

Mathuranathan — I can't spot any mistake. Please correct if I am wrong.

Log in with:

SEARCH ARTICLES

Search...

Go

LIKE THIS WEBSITE ? SUBSCRIBE

Enter your email address to subscribe to this blog and receive notifications of new posts by email.

Join 109 other subscribers

Email Address

Subscribe

PARTICIPATE AND WIN

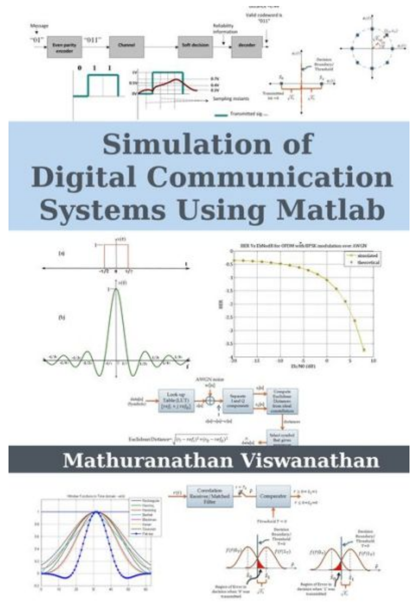
Participate in Forums and win 32-GB microSD every month. Just top the leaderboard with maximum reps at the end of each month

SEARCH QUESTIONS

Search questions & answers

Search

GRAB THIS EBOOK TODAY



RECENT QUESTIONS

[BER for BPSK modulation in AWGN, Rayleigh, and Rician fading channels]

1 Answer | 0 Votes

[OFDM with QPSK]

0 Answers | 0 Votes

[QPSK Complex input]

1 Answer | 0 Votes

Correct the statement

1 Answer | 0 Votes

Could not able to get the results,can u help me in this concern

1 Answer | 0 Votes

RECENTLY SUBMITTED

A mathematical theory of communication – Shannon

An Introduction to Information Theory and Applications

More tutorials

CATEGORIES