ECOLE POLYTECHNIQUE FEDERALE DE LAUSANNE

School of Computer and Communication Sciences

Software-Defined Radio:	Solutions to Assignment 1
A Hands-On Course	Date: 21-28 September 2016

Exercise 1.

1. Let N be the length of the vector `s` containing the samples. The command to generate vector `t` is

   ```
   t = linspace(0, (N-1)/fs, N);
   ```

   Equivalently, we can do:

   ```
   t = (0 : 1/fs : (N-1)/fs);
   ```

2. From the discussion in the class, we know that the points of the DFT are spaced $1/T_p$ Hz apart, i.e., $1/(NT_s) = f_s/N$. Notice however that when we compute the FFT we can pass as extra parameter the blocklength `NFFT` that Matlab will use. For efficiency, the value of `NFFT` should be a power of 2 greater or equal to N. This is accomplished by `NFFT = 2^nextpow2(`N`)`. Matlab will automatically do the necessary padding with zeros.

   If `s_f=fft(s,NFFT)` and `NFFT` is even, then the components of `s_f` are associated to the frequencies

   $$\underbrace{0, \frac{f_s}{\texttt{NFFT}}, \ldots, \frac{f_s}{2} - \frac{f_s}{\texttt{NFFT}}}_{\texttt{NFFT}/2 \text{ components}}, \underbrace{\frac{f_s}{2}, -\frac{f_s}{2} + \frac{f_s}{\texttt{NFFT}}, \ldots, -\frac{f_s}{\texttt{NFFT}}}_{\texttt{NFFT}/2 \text{ components}}.$$

   If we plot `fftshift(s_f)` then the components are associated to the frequencies $\frac{f_s}{2}, -\frac{f_s}{2} + \frac{f_s}{\texttt{NFFT}}, -\frac{f_s}{2} + 2\frac{f_s}{\texttt{NFFT}}, \ldots, 0, \frac{f_s}{\texttt{NFFT}}, \ldots, \frac{f_s}{2} - \frac{f_s}{\texttt{NFFT}}$. Notice that for the first component we could have written $-\frac{f_s}{2}$ rather than $\frac{f_s}{2}$, because the spectrum vanishes at both frequencies. Thus one possibility to create vector `f` is

   ```
   f = linspace(-fs/2, fs/2-fs/NFFT, NFFT);
   ```

   Another possibility is

   ```
   f = (-fs/2 : fs/NFFT : fs/2-fs/NFFT);
   ```

Exercise 2. *Note:* Since `title` is the name of a built-in MATLAB function, you cannot use it as the name of a function argument. In the following code, we called the argument `plottitle` instead.

```
% TFPLOT Time and frequency plot
%    TFPLOT(S, FS, NAME, TITLE) displays a figure window with two subplots.
```

```
%    Above, the signal S is plotted in time domain; below, the signal is plotted
%    in frequency domain. NAME is the "name" of the signal, e.g., if NAME is
%    's', then the labels on the y-axes will be 's(t)' and '|s_F(f)|',
%    respectively.  TITLE is the title that will appear above the two plots.

function [] = tfplot(s, fs, name, plottitle)

% Note: Since TITLE is the name of a built-in Matlab function,
% you cannot use it as the name of a function argument.
% In the following code, we called the argument PLOTTITLE instead

% Compute the time and frequency scales
t = linspace(0, (length(s)-1) / fs, length(s));

NFFT = 2^nextpow2(length(s));
f = linspace(-fs/2, fs/2-fs/NFFT, NFFT);

% compute the FFT
s_f = fft(s,NFFT);

figure;
% First plot: time
subplot(2,1,1); plot(t, s); % stem(t, s,'.');
xlabel('t [s]'); ylabel(sprintf('%s(t)', name));
title(plottitle);

% Second plot: frequency
% We use fftshift to move the coefficients for negative frequencies to the left
subplot(2,1,2); plot(f, fftshift(abs(s_f))); % stem(f, fftshift(abs(s_f)),'.');
xlabel('f [Hz]'); ylabel(sprintf('|%s_F(f)|', name));
```

Exercise 3. Please see the provided Matlab routines, which are self-explanatory.

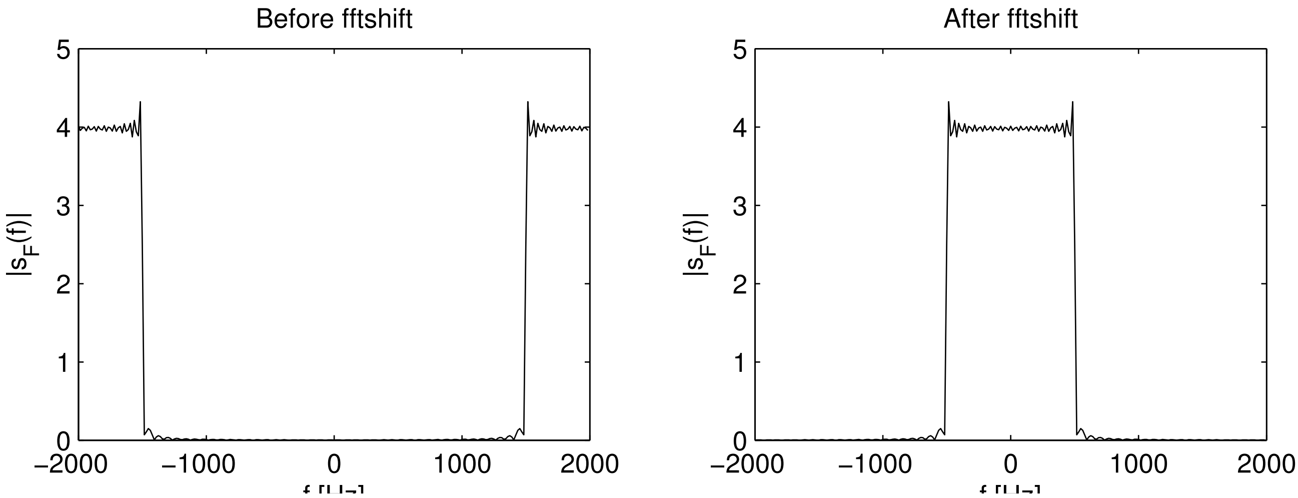

Figure 1: Effect of the `fftshift` command.