Homework #5, EECS 451, W04. Due **Fri. Feb. 20**, in class

1. [B 10] Concept(s): ***Block diagrams from system specifications.***
In class the block diagram for a digital sine-wave generator was developed: a system whose impulse response is $h[n] = \cos(\omega_0 n + \phi)\, u[n]$. An alternative approach would be to use a system whose *step response* is $\cos(\omega_0 n + \phi)\, u[n]$.
(a) [10] Draw the Direct Form II implementation of this alternative system.
(b) [0] Which digital sine-wave generator is better, the one developed in class that is driven by an impulse, or the one developed here that is driven by a step function?
If you are struggling with this, you may consider the specific case where $\omega_0 = \pi/2$ and $\phi = 0$.

2. [B 20] Concept(s): ***Steady-state response and frequency response.***
The signal $\cos(\omega_0 n)\, u[n]$ is applied to a filter with impulse response $h[n] = (-1/2)^n\, u[n]$.
(a) [10] Use PFE to find the output signal $y[n]$.
(b) [5] Plot the amplitude of the steady-state response as a function of ω_0.
(c) [5] Plot the magnitude response of the filter.
(d) [0] Discuss.

3. [U 10] Concept(s): ***DTFS analysis.***
(a) [5] Find the DTFS coefficients of the following signal: $x[n] = \cos(\pi n) + \sin\left(\frac{2\pi}{3} n\right)$.
(b) [5] Plot (by hand or use MATLAB) the power density spectrum and phase spectrum of $x[n]$.

4. [B 0] Concept(s): ***DTFS***
A discrete-time "impulse-train" signal is defined as follows: $s_M[n] = \sum_{m=-\infty}^{\infty} \delta[n - mM]$.
(a) [0] Sketch the particular signal $s_4[n]$. Your graph should at least cover the range $-6 \le n \le 10$.
(b) [0] Find the DTFS representation of a general impulse-train signal $s_M(n)$ for arbitrary $M \in \mathbb{N}$.

5. [G 10] Concept(s): ***DTFS and downsampling***
Consider the "zeroing" form of downsampling by a factor of M, defined by

$$y[n] = \begin{cases} x[n], & n \text{ a multiple of } M \\ 0, & \text{otherwise.} \end{cases}$$

This input-output relationship defines a linear, but time-varying, DT system. Nevertheless, in this case we can still find a relationship between $Y(z)$ and $X(z)$. Derive the following general expression for $Y(z)$ in terms of $X(z)$:

$$Y(z) = \frac{1}{M} \sum_{k=0}^{M-1} X\left(e^{-\jmath \frac{2\pi}{M} k}\, z\right).$$

Hints: $y[n] = x[n]\, s_M[n]$, where $s_M[n]$ is defined in previous problem. The case $M = 2$ was derived in lecture.

6. [B 30] Concept(s): ***DTFS implementation.***
For reasons that will become clear in Chapter 6, MATLAB does not have a `dtfs` command, *i.e.*, it does not have a built-in command for computing the DTFS coefficients of a periodic signal. (Try `lookfor dtfs`.) Your company wants to display the power density spectrum of periodic violin signals (see below), so you need to write a MATLAB function `dtfs1`. Your function gets one period of the signal's values as input, *i.e.*, $x[0], x[1], \ldots, x[N-1]$, stored in a vector `xv` of length N, and returns the DTFS coefficients $c_0, c_1, \ldots, c_{N-1}$, stored in a vector `cv` of length N.
Fortuitously, a former employee started this project and has given you a partially completed MATLAB m-file `dtfs1_template.m` (available from the usual `mfiles` directory on the web site). However, this employee did not quite finish the project (hence "former" employee...).
Copy the template function to your directory and rename it `dtfs1.m`, and then edit it and complete the missing lines to compute the DTFS coefficients. If you are proficient in MATLAB, then your mfile should only have one loop, not two! Test your m-file using (for example) the signal in Example 4.2.1(c) on p. 249 of your text. When you type

```
>> xv = [1 1 0 0]
>> cv = dtfs1(xv)
```

MATLAB should return something like the following (if your function is working correctly):

```
cv =
0.50
0.25 - 0.25i
0 - 0.00i
0.25 + 0.25i
```

(a) [20] Turn in a printout of your `dtfs1.m` file.

(b) [10] Use your `dtfs1` function to compute the DTFS coefficients of this periodic signal: $\{\ldots, \underline{2}, 1, 0, 0, 0, 1, \ldots\}$.

7. [B 45] Concept(s): ***Spectrum of sampled periodic signals via DTFS.***
Violin signals are approximately sawtooth functions:

$$x(t) = 10\left(\frac{(t \bmod T_0)}{T_0} - \frac{1}{2}\right) = \sum_{k=-\infty}^{\infty} \tilde{x}_a(t - kT_0), \quad \tilde{x}(t) = \begin{cases} 10\left(\frac{t}{T_0} - \frac{1}{2}\right), & 0 \le t < T_0 \\ 0, & \text{otherwise.} \end{cases}$$

Suppose we form a DT signal $x[n]$ by sampling the above violin signal (having a fundamental frequency $F_0 = 1/T_0 = 800\text{Hz}$) at a sampling rate of $F_s = 8000\text{Hz}$.

(a) [10] Find $x[n]$ and determine its period. Hint: $x[n] = x(nT_s)$ where $T_s = 1/F_s$. See `help mod` in MATLAB if you are unfamiliar with the *modulo* function. If you are still unsure about the definition $x(t)$, here is MATLAB code that plots it.

```
% violin1g.m, plot violin signal
xa = inline('10 * (mod(t, To)/To-0.5)', 't', 'To');
To = 1/800;
t = linspace(0,4*To,101);
plot(1000*t, xa(t,To), '-')
xlabel 'Time [milliseconds]', ylabel 'x(t)'
title 'Violin signal (sawtooth wave)'
```

(b) [5] Is there any aliasing at this sampling rate?

(c) [30] Use MATLAB to plot the power density spectrum of $x[n]$, after using your `dtfs1` function to compute the DTFS coefficients. However, for the horizontal axis of your plot, display the *continuous-time* frequencies (in Hz) rather than the digital frequencies ω_k. (This is how digital spectrum analyzers work.)
Hint: recall from Chapter 1 that the relationship between any digital frequency $\omega \in [-\pi, \pi]$ and the corresponding continuous-time frequency F of a sinusoidal signal is $\omega = 2\pi F/F_s$, or, rearranging, $F = \frac{\omega}{2\pi} F_s$. Note that any F computed properly will lie in the interval $[-F_s/2, F_s/2]$.

8. [B 15] Concept(s): ***DTFT analysis.***

   (a) [5] Find the DTFT $X(\omega)$ of the following signal:

$$x[n] = \left\{ \underline{1}, 1, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}, \ldots \right\}.$$

   (b) [10] Use MATLAB to plot the magnitude and phase spectra. For this problem (only), make ω go from -3π to 3π in these plots. As always, clearly label your plots and turn in your m-file too. To put ω's in MATLAB axis labels, use `xlabel '\omega '`.

9. [B 10] Concept(s): ***Inverse DTFT.***

   Find the signal having the following spectrum.