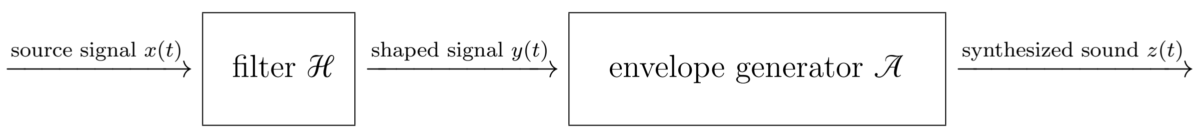
UCLA — Electrical Engineering Dept.
EE102: Systems and Signals
Project: A Simplified Music Synthesis
Due: 12/12/2014

---

1. Please form groups of *up to* three people (groups of one or two people are OK). Make sure that at least one person in the group is reasonably familiar with MATLAB.
2. Generate MATLAB code and all the figures. Please, comment your code! (Remember that in MATLAB anything that follows a % sign is considered a comment.)
3. Each group should also generate one *short* report. You can use any word processor. Even a simple text file will do. A PDF output file is preferred.
4. The minimum requirements are: the short report, figures, MATLAB code. If you cannot embed the figures in your document, please zip them with your report.
5. Upload your project to CCLE, only one per group. Make sure the names of all the group members are clearly indicated.

Project Description

In this project, we demonstrate the applications of EE102 in music synthesis, that is, how a machine can generate music. One of the methods to synthesize a sound is illustrated by the following block diagram:



The source signal $x(t)$, is a harmonically rich sound, in that it contains various frequency components. For example, a plucked string vibrates with a certain set of frequencies, which are determined by the tension and the length of the string. The lowest frequency is known as the *fundamental frequency* (f_0), or the *pitch*. The string also vibrates at frequencies that are integer multiples of f_0, and such set of frequencies are called *harmonics*. Thus, the source signal can be modeled as a sum of sinusoidal waves whose frequencies are integer multiples of f_0.

The body of an instrument acts as a filter by resonating the source signal. A filter weights each frequency component as defined in its transfer function. As a result, some of the frequency components become louder while others become softer. The magnitude difference between the harmonics determines the *timbre* of the sound. The resonator can be modeled as a linear, time-invariant and causal system $\mathscr{H}$. The source signal $x(t)$ is applied to the system $\mathscr{H}$ to produce the shaped signal $y(t)$.

Then a time domain manipulation is performed by an envelope generator to imitate sound from a real instrument whose loudness changes over time. For example, the sound from a plucked string decays very fast. In this project, we are trying to build a simple model to generate a sound.

Questions

1. We will first analyze given sounds from real instruments. Plot and analyze spectra from different musical instruments with the same note (A4). Run `analysis.m` to plot the time domain signal

and frequency domain spectrum. Use `soundsc` to listen to the loaded sound.

(a) Compare the time domain signals in terms of their fundamental periods ($T_0 = 1/f_0$), their shapes in one period, and the overall envelopes.

(b) Compare the frequency domain spectra in terms of the frequency components with high magnitude and their relative magnitudes.

(c) What are the main factors that determine the pitch and the timbre of a sound? Explain based on the observations made in (a) and (b).

2. Now we will generate a source signal with a single tone. Generate a sinusoidal wave $x_0(t)$ with the fundamental frequency 440 Hz and listen to the sound by completing the following MATLAB source code. Assume the sampling frequency (f_s) is 22 050 Hz.

```
% ———— parameter definition ——
Fs = 22050;     % sampling frequency [Hz]
F0 = 440;       % fundamental frequency [Hz]
D  = 1;         % duration in [sec]

% ———— generate signal ————
t = linspace(1/Fs, D, Fs*D);
x0 = sin( ... );

% ———— play sound ————————
soundsc(x0, Fs);

% ———— plot ——————————————
plot_spectrum(x, Fs, 'single tone')
```

(a) What is the fundamental period in seconds? Verify it with the generated sinusoidal signal.

(b) Change f_0 to a value you want. How does the sound and the plots change? Explain your observations in terms of pitch and timbre.

3. Since $x_0(t)$ has only one frequency component, it is not sufficient to make a rich sound. Now we will generate a harmonically rich source signal $x(t)$, which is defined as

$$x(t) = \sum_{k=1}^{N} \sin(2\pi k f_0 t)$$

where N is the number of harmonics. Use the same fundamental frequency and sampling frequency given in 2. Listen to it and analyze it with `plot_spectrum` function.

(a) The *sampling theorem* states that for a perfect reconstruction, the sampling frequency should be at least twice larger than the maximum frequency of the signal. That is $f_s > 2B$ is required, where B is the maximum frequency of the band-limited signal. Based on this theorem, what is the maximum value of N?

(b) Compare the sound and the plots to those from $x_0(t)$ generated in 2. Explain your observations in terms of pitch and timbre.

4. We then model a resonator of a virtual instrument as a filter defined by a simple second-order differential equation:

$$A_1 \frac{\mathrm{d}^2 y(t)}{\mathrm{d}t^2} + A_2 \frac{\mathrm{d}y(t)}{\mathrm{d}t} + A_3 y(t) = B_1 \frac{\mathrm{d}x(t)}{\mathrm{d}t} + B_2 x(t), \quad t \geq 0,$$

where $A_1 = 1, A_2 = 2 \times 10^3, A_3 = 1 \times 10^6$, $B_1 = 5 \times 10^3$, and $B_2 = 1 \times 10^8$. We denote this system by $\mathscr{H}$. Here, t is in seconds.

(a) Use MATLAB to compute and plot the impulse response $h(t)$ and the unit step response $g(t)$ of $\mathscr{H}$ by completing the following source code.

```
Fs = 22050;
Ts = 1/Fs;
t  = [0:Ts:0.1];

num = (...);
den = (...);
sys = tf(num, den);

h = impulse(sys, t);
g = step(sys,t);

figure
subplot(2,1,1)
plot_spectrum(h, Fs, 'impulse response')
subplot(2,1,2)
plot_spectrum(g, Fs, 'step response')
```

(b) Compute the filter output $y(t)$ from the input signal $x(t)$ generated in 3 with MATLAB . Compare the sound and the plots to those from $x(t)$. Explain your observations in terms of pitch and timbre.

5. Now an envelope $a(t)$ will be multiplied to $y(t)$ in time domain in order to make the sound more realistic. An exponential model can be used to imitate sound from a plucked string which decays very fast. That is,

$$a(t) = \mathrm{e}^{-5t}.$$

(a) Perform the time domain envelope shaping with the given exponential model. Compare the sound and the plots to those from $y(t)$. Explain your observations in terms of pitch, timbre, and loudness change in time.

(b) (optional) A low-pass filtering is often used to attenuate undesirable high-frequency components. Apply a low-pass filter with the following MATLAB source code and explain your observations.

```
Fc = 5000; % cut-off frequency in [Hz]

[b,a] = butter(9, Fc/Fs);
z = filter(b,a,z);
plot_spectrum(z, Fs, 'zl(t) : low-pass filtered signal');
```

6. Create your own musical piece with synthesized sound (about 10 seconds long). Feel free to use your own transfer function and/or time domain envelope function. Some of the fundamental frequencies for notes are given in `notes.m`.

   (a) Plot the resulting music using `plot` function in time domain only.

   (b) Describe the methods you used and your reasons briefly.