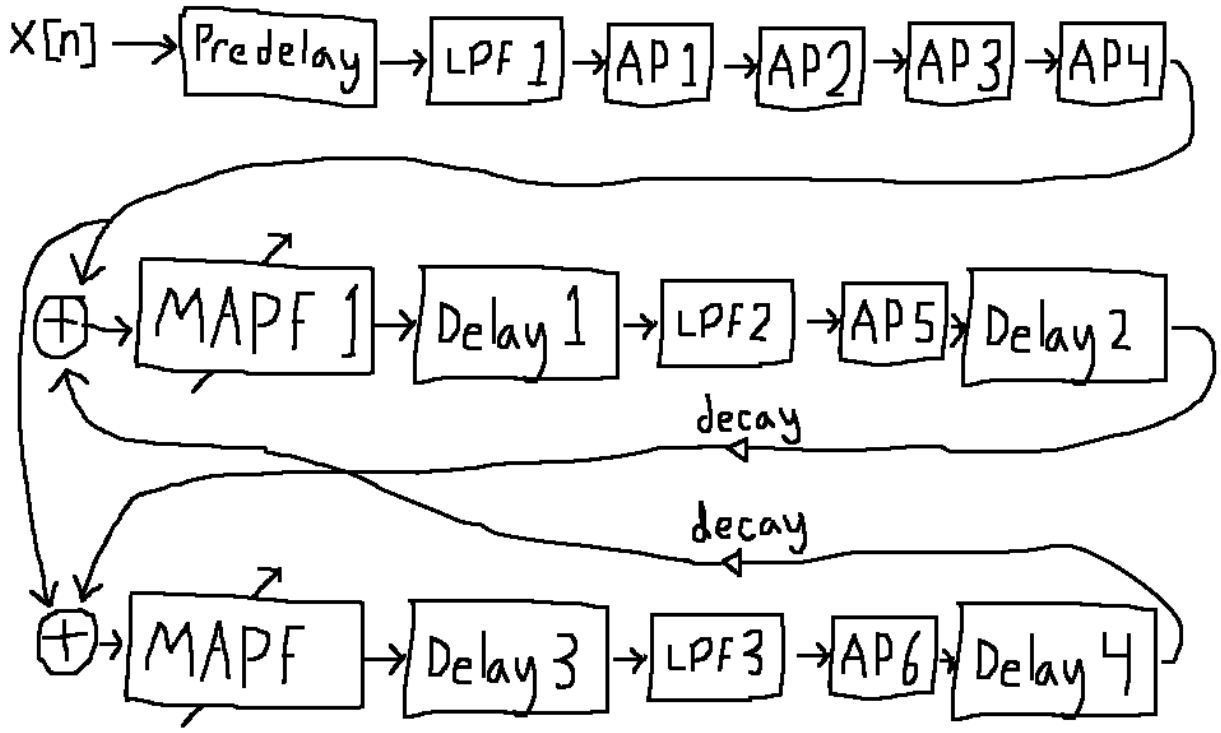
DST Homework #3: DUE MARCH 1st at 11:59PM

Please submit your assignment to Brightspace as a .zip file containing the following:

- a jupyter notebook containing your solution to the programming question
- the original “snare.wav” file

---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ----

Read pages 660-666 of the attached “dattorro.pdf” for background and then implement the following mono-in, stereo-out plate reverb derived from the paper. Test it using the provided “snare.wav”



Parameters:

APF1: D=210, g=0.75
APF2: D=158, g=0.75
APF3: D=561, g=0.625
APF4: D=410, g=0.625

MAPF1: D=1343 +/- 12, g=0.7
Delay1: D=6241
APF5: D=3931, g=0.5
Delay2: 4681

MAPF2: D=995 +/- 12, g=0.7
Delay3: D=6590
APF6: D=2664, g=0.5
Delay4: D=5505

Choose Pre-delay, LFO freq, LPF and decay parameters by ear.

Output: Note that there is no single output node on the block diagram. The left and right channel outputs are derived by combining outputs from various points in the delay lines throughout the reverb according to the following formulas. You will need a way to grab samples from the middle of the delay lines. For example delay[394] means that you will grab the data 394 samples into the delay line (394 samples behind the write pointer if using a circular buffer). This is referred to as tapping a delay line.

y_L = delay1[394] + delay1[4401] – apf5[2831] + delay2[2954] – delay3[2945] – apf6[277] - delay4[1066]

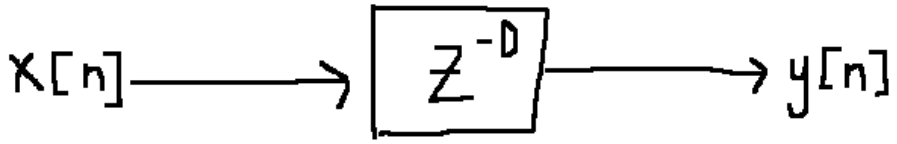
y_R = delay3[522] + delay3[5368] – apf6[1817] + delay4[3956] – delay1[3124] – apf5[496] - delay2[179]

---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ----

NOTES:

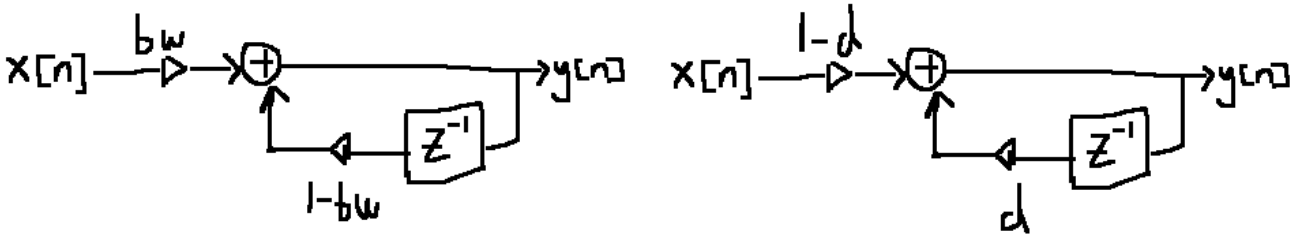
Because of the modularity and relative complexity of this reverb’s design, it would be incredibly difficult to derive a short set of difference equations for the entire system as we have done with smaller systems. I recommend taking an object-based approach to its implementation. You may find it useful to create classes for each of the reverb’s building blocks:

Delay Line:



fixed length delay line. Should include a “tap” method which will return the sample at point **n** in the delay line (ie. **n** samples behind the write pointer).

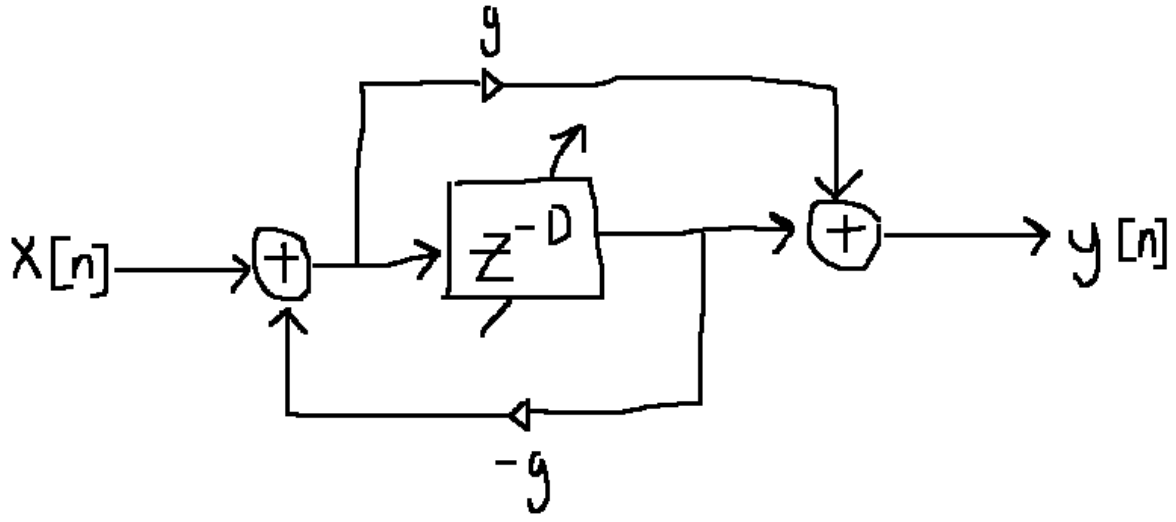
One Pole Lowpass Filter:



Dattoro’s paper includes two different versions of this filter: one labelled with “bandwidth” for the input stage, and one labelled with “damping” for the tanks. The only difference between these filters is that the effect of the control variable (bandwidth or damping) is inverted. Feel free to implement both versions or just one.

Modulated Allpass: delaying allpass filter with moveable delay time.

Use linear or allpass interpolation. Include methods for setting a new delay length and returning a sample at point **n** in the delay line. You can use this for your fixed-length allpasses as well or create a separate class for those (you could make them slightly more efficient than the modulated all-pass).



Once you have created these classes, connect them together in the correct arrangement and tap your stereo outputs from the reverb.

Test different values for damping and bandwidth as well as parameters for your LFOs for a smooth and natural sounding reverb. Slow and subtle LFOs will likely sound the best; you just want them to increase diffusion, not cause a chorus effect. Set slightly different frequencies for each LFO to reduce the chances of your all-passes pulsing together.

For a short input like the given snare drum, you will want to extend your output's length to allow time for the reverb to ring out. The easiest way to do this would be to add a few seconds of silence (np.zeros, etc) to the end of the input signal before processing.

Once your reverb is working, it can be helpful to mix some dry signal back in at the end because the above algorithm will give a fully wet sound.

The included "example_reverb.wav" was created by passing the provided "snare.wav" through an implementation of this plate reverb and can be used as a reference for what yours should be sounding like. It wasn't tuned very carefully so you may be able to get yours sounding better.