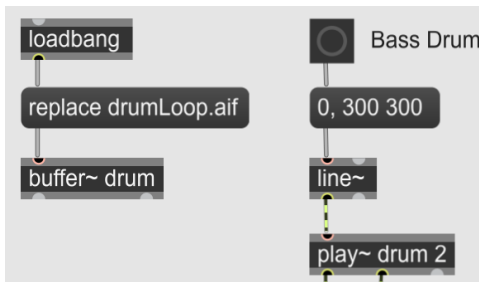


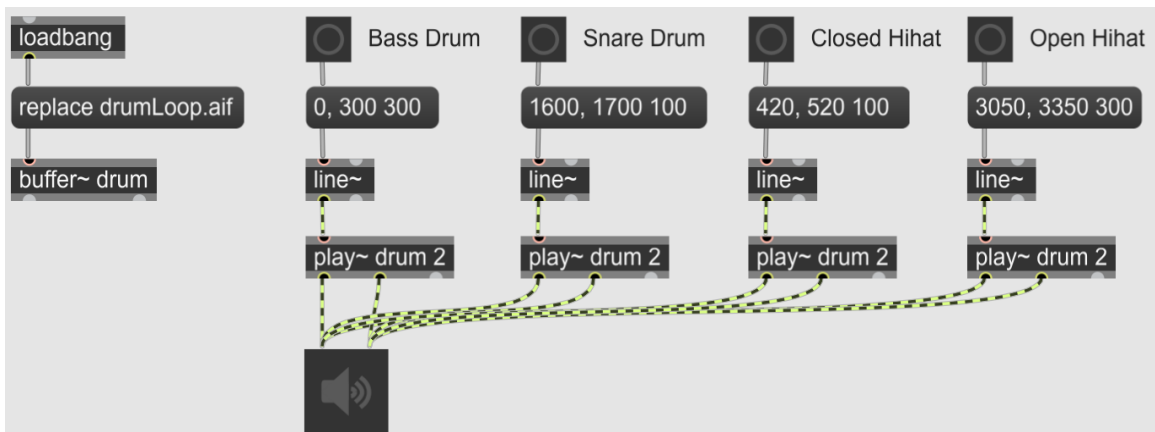
Lecture 19: Sampling & Delay

Example 1: Sampling ("1_sampling.maxpat")

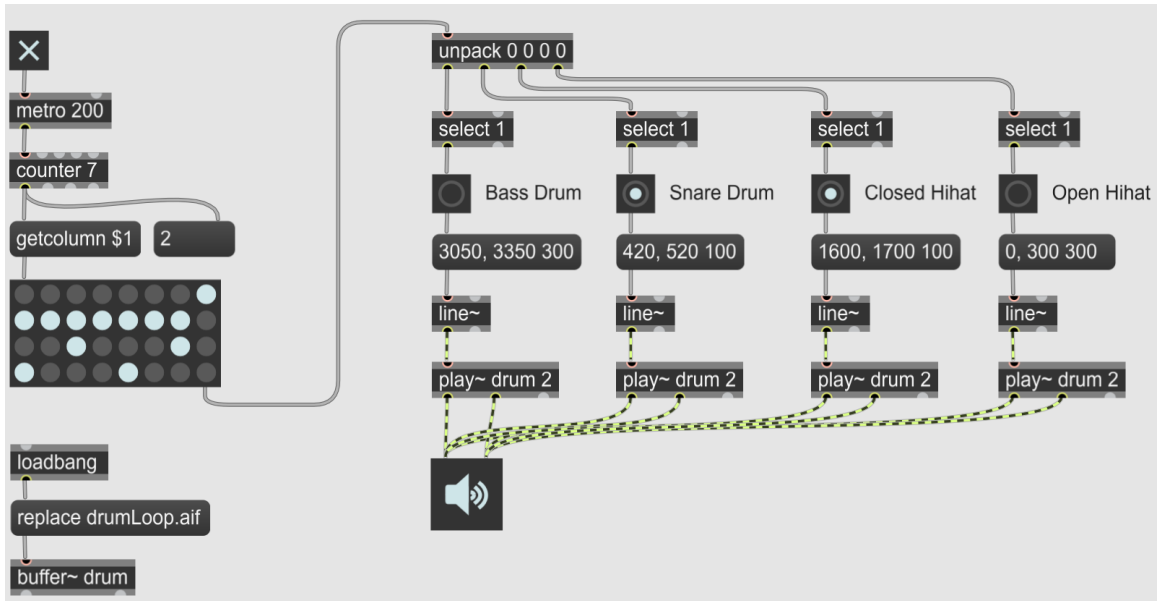
- Use the "line~" object to play part of the sample. The example below will play the first 300ms of the audio, which contains a bass drum sound.



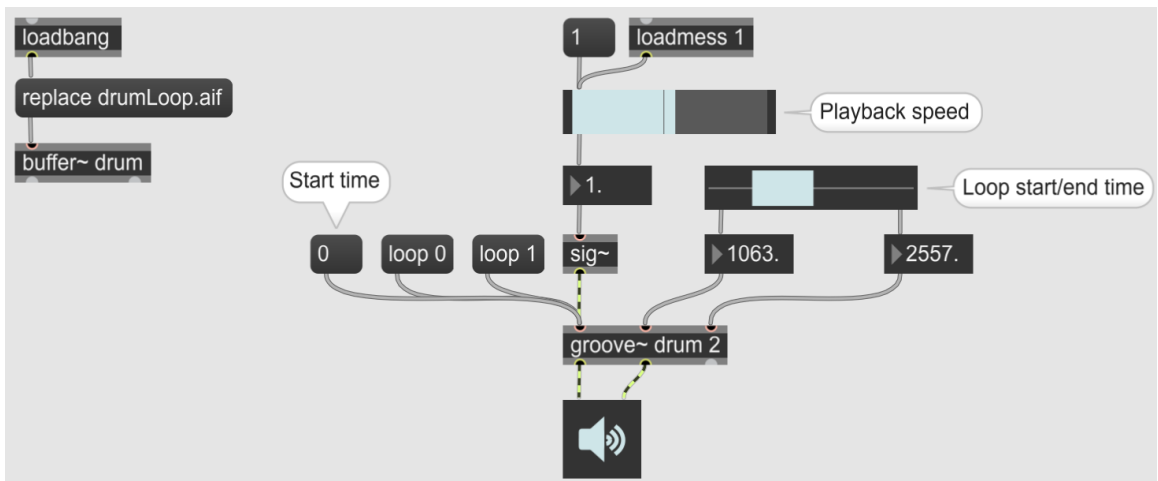
- By sampling different parts of the audio, we can create a sampler with different drum sounds.



- Use the “matrixctrl” object to create a sampler-based drum machine like what we did previously.



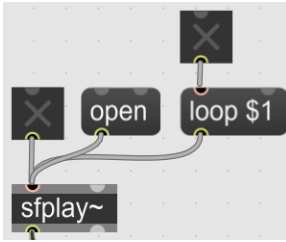
- Use the “groove~” object for variable-rate looping, which is helpful in making a looper with some audio sample.



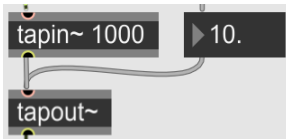
- Messages “loop 0” and “loop 1” controls the loop behavior.
- The input signal (created by the “sig~” object in this example) determines the playback speed.
- The second and third inlets determine the start and end time of the loop.
- The number after “groove~ drum” (2 in this example) determines the number of channels.

Example 2: Delay ("2_delay.maxpat")

- Use "sfplay~" to play an audio stored on the machine.

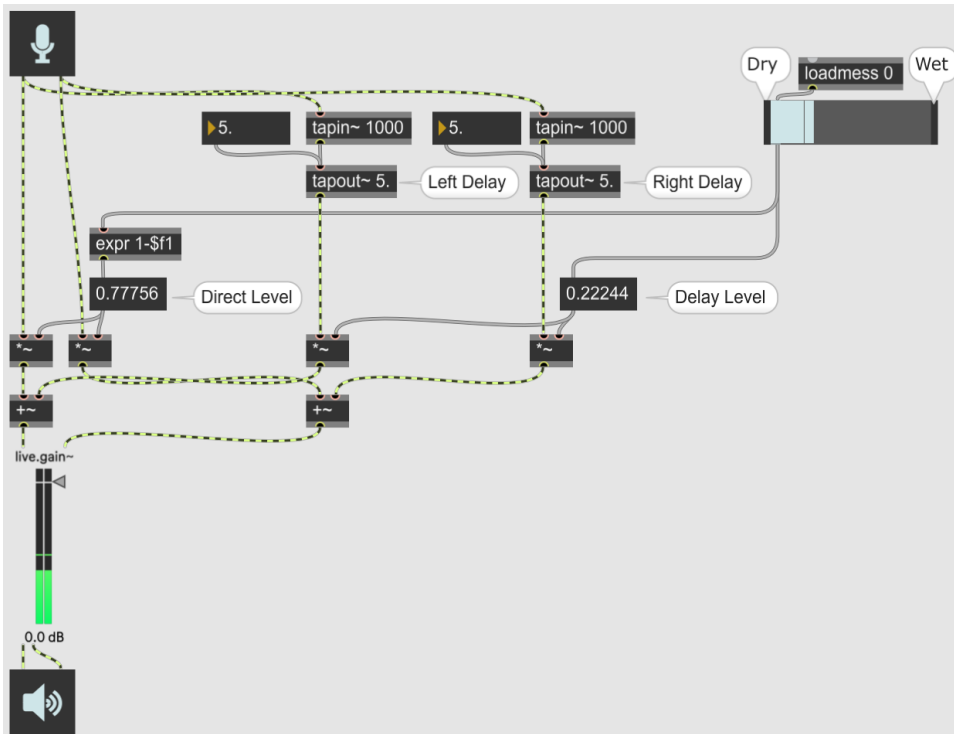


- A 0/1 value set the playback, which is controlled by a toggle in this example.
 - Sending an "open" message will prompt the user to select an audio file through the file explorer.
 - The "loop 0" ("loop 1") message enables (disables) the looping behavior.
- Use the "tapin~" and "tapout~" to create a delayed version of the original signal.

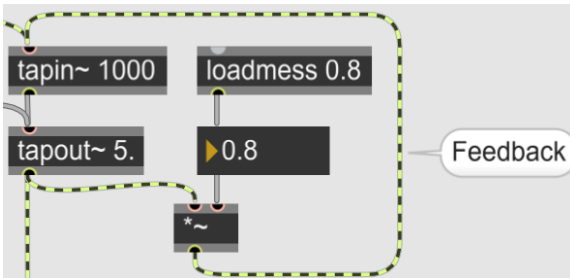


- Note that the cord between the "tapin~" and "tapout~" objects is gray rather than yellow, meaning that there's no signal passing between them. Instead, the cord is to indicate that the two objects share the same delay memory.

- In the second example, we can create a similar effect using stereo inputs from the microphone.

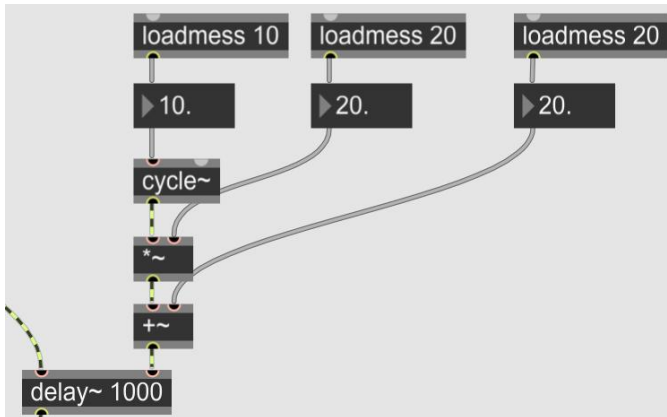


- In the third example, we can create a feedback loop on top of the delay effect to create an “infinite echo” effect, where the input audio signal keeps being delayed with decayed amplitude.



- In this example, a decayed version (80% of the original) of the delayed signal will be sent back to be delayed again (and again and again...until it becomes inaudible).

- Alternatively, we can use the “delay~” object to create the delay effect, which also allows signal input for the delay time.



- This allows us to create the “flanging” effect, which is created by mixing the input signal with a delayed version of the input signal where the delay time is oscillating (a sine wave in this example).

