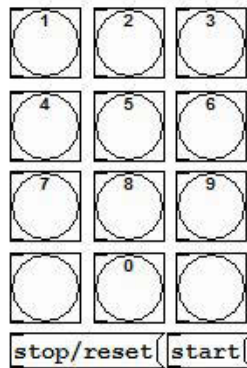
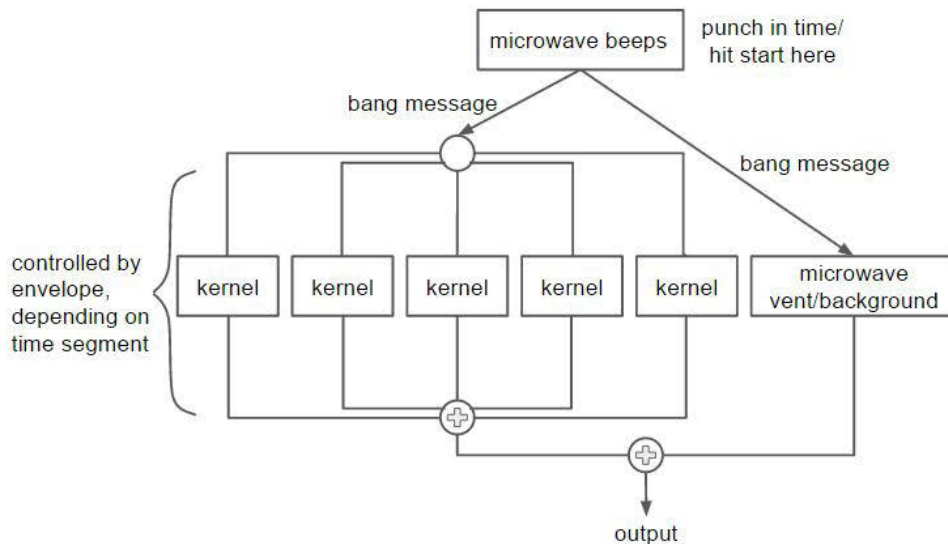


My initial focus for the project is the popcorn kernels, the background microwave and ventilation, and the time-setting and starting beeps of the microwave. I think that for the actual user interface for this project, it would be humorous to have something that looks like a microwave. So having the user press numbered buttons to choose how long to pop the popcorn for (or maybe just the popcorn button that some microwave have) and then press "start." I think I can make these buttons make noise instantly, as with an actual microwave. Based on the time punched in, the microwave background noises would last for that long. After the start, the microwave humming would be the only sound for the first 30 seconds or so, and then the popcorn would begin to pop slowly. The popping then should speed up, and after about 2 minutes should ramp down and eventually stop entirely, and the final sound should be the microwave beeping that it is done. There also needs to be a reset button so that this process can be done multiple times without having to re-open the patch.

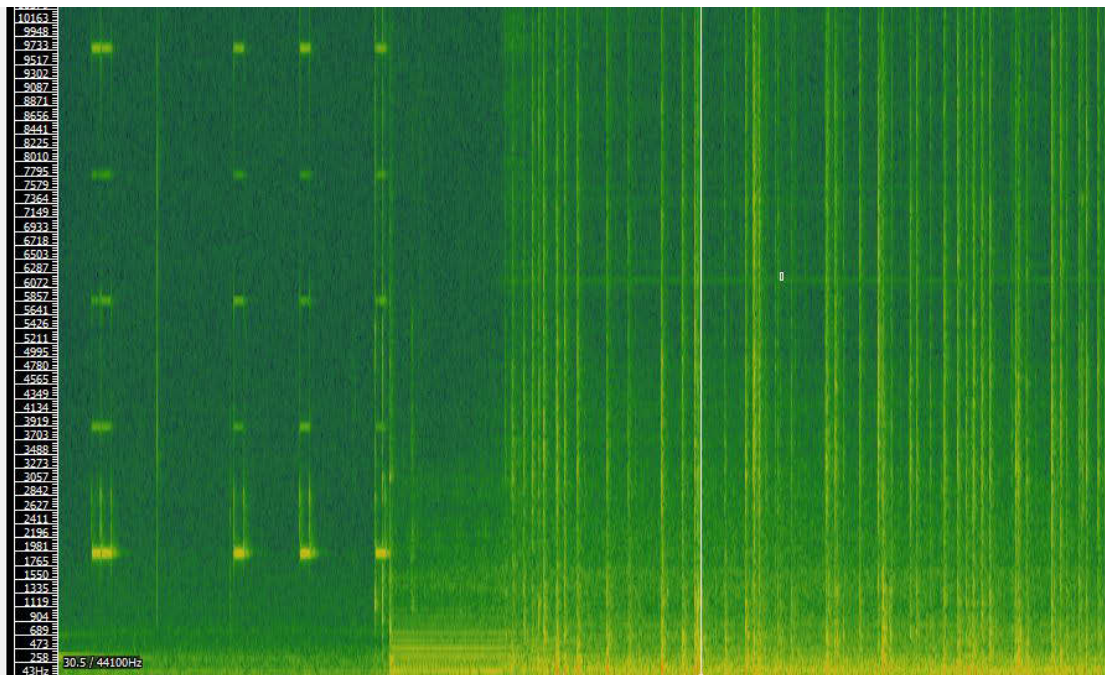
Here is what the simplified UI could look like:



Here is a block diagram with the flow being an odd combination of time and signals that I will figure out eventually. Kernel is an abstraction that outputs the sound of a single kernel popping. Microwave vent/background could be a subpatch, and the beeps should also be abstractions.



Sonic Visualiser Screenshot: This is of the 30 second clip presented in class for FP1.



Kernels Popping:

The kernel pops appear on Sonic Visualiser to mostly be an explosion of noise. The burst lasts for less than 43 ms, and have a very fast attack with a slightly more gentle decay, although the entire thing is pretty quick.

To get the timing of the actual kernels popping, I will split the process into three parts: the ramp up, the steady popping, and the ramp down. And to make sure that multiple kernels can be heard popping at once without cutting each other's envelope/sound off, I will have five different kernels that can be randomly popped. When starting to pop, the kernels go off at about 1 every 1.5 seconds. Ramping down is even slower; the kernels end up popping about once every 2 seconds and, after about 5 last pops, they stop entirely. In the middle time window, there are 3 or 4 kernels popping every second. I can use a random number generator that generates a number from 0 to $n-1$, where $n-1$ is the number of milliseconds it takes for whichever popcorn phase to produce a single popped kernel (except I think it should be $(n-1)*5$ so that five different kernel abstractions can be used, as I explained above. This might change as time goes on.). The kernel abstractions I write can then accept a 0 as a trigger to set off a pop, and all other numbers will not yield noise.

Microwave Beeps:

The beeping sounds of the buttons have a very spaced out spectrum, and the frequencies do not seem very well defined. The six main frequency bands are centered around 1937, 3919, 5900, 7795, 9800, and 11750 Hz; the width of each band is about 130 to 150 Hz. The beeps last about 150 ms, and should be fairly easy model with some band pass filtered noise and an envelope.

MIT OpenCourseWare
<http://ocw.mit.edu>

21M.380 Music and Technology: Sound Design
Spring 2016

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.